

Quiz[?] CREATOR

4. semesterprojekt
Dokumentation, Gruppe 3
28/5 2015



Underskrifter

Dato for aflevering: **28/05/2015**

Gruppemedlemmer:

Anders Bæk Møller	201270727
-------------------	-----------

Bjørn Sørensen	201270432
----------------	-----------

Jesper Christensen	201270478
--------------------	-----------

Lasse Beck Thostrup	201271288
---------------------	-----------

Lars Harup Holm	201370744
-----------------	-----------

Loc Dai Le	201371016
------------	-----------

Michael Toft Jensen	201371180
---------------------	-----------

Vejleder: Frank Bodholdt Jakobsen

Indholdsfortegnelse

Kapitel 1	Ordliste	5
Kapitel 2	Indledning	7
Kapitel 3	Krav	9
3.1	Aktører	9
3.1.1	Aktør-beskrivelse	10
3.2	Use Case-beskrivelse	11
3.3	Ikke-funktionelle krav	19
Kapitel 4	Forundersøgelse	21
4.1	Grafisk brugerflade-teknologier	21
4.2	Database-teknologier	21
4.3	Systemarkitektur	22
Kapitel 5	Systemarkitektur	23
5.1	Domænemodel	23
5.2	Logical View	23
5.3	Implementation View	24
5.4	Data View	26
5.5	Deployment View	27
Kapitel 6	Design	29
6.1	QuizModel	29
6.2	Sekvensdiagrammer	29
Kapitel 7	Implementering	41
7.1	Klassediagrammer	41
7.2	Detaljeret beskrivelse af systemet med Doxygen	41
7.3	Beskrivelse af views	42
7.3.1	Quiz views	42
7.3.2	Group views	48
7.3.3	User views	50
7.3.4	Favorite views	52
7.4	Remote Validation	53
Kapitel 8	Test	55
Kapitel 9	Accepttest-specifikation	57
Kapitel 10	Bilags-CD indhold	73

Appendiks A Domænemodel	75
Litteratur	77

AASH Antal af samtidige hændelser (Use Case)

ASE Aarhus School of Engineering

DAL Data Access Layer

EF Entity Framework. Et .NET-framework til databasehåndtering

GUI Graphical User Interface

UC Use Case

Session En aktiv forbindelse mellem brugerens webbrowser og applikationen QuizCreator.

Tilgås applikationen fra endnu en fane, har brugeren altså 2 sessioner åbne. (Use Case)

2

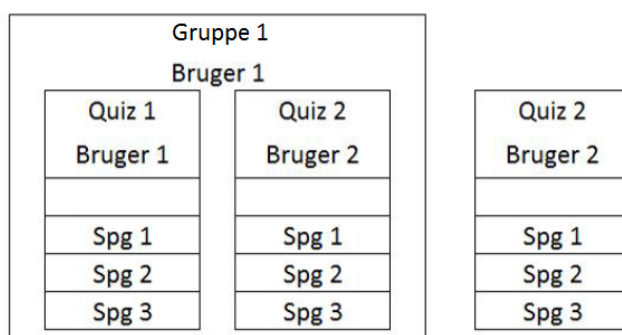
Indledning

Tanken med projektet er at opbygge en webapplikation, som vil gøre det let for mennesker imellem at teste hinandens viden. Tests bruges alle steder, både i konkurrencer men også i faglige sammenhænge.

Meningen med applikationen er at man vil kunne bruge den uanset om man er alene og søger at teste viden indenfor et emne, eller om man er en gruppe af mennesker som søger underholdning til en aften. Første prioritet med applikationen er, at den er let og intuitiv at navigere og benytte om enten man er en 4. klasses elev eller IT studerende på en videregående uddannelse. Brugere af applikationen skal være i stand til at kunne logge sig ind og oprette en "gruppe" (hvorved de bliver administrator af gruppen), som andre kan tilmelde sig. Der ud over skal man kunne oprette sin egen quiz, og herefter tilføje en række spørgsmål som brugeren selv opretter.

Hvis brugeren ikke selv vil lave quizzer skal han kunne søge i en database, som indeholder alle tidligere oprettede quizzer. Dette gøres ved at hver quiz får såkaldte "tags" - f.eks. kan man oprette en quiz om division, og give det tagget "division" og "matematik". En quiz kan således ligge frit tilgængeligt, uden at være tilknyttet en gruppe. Bruger man webapplikationen uden at være logget ind kan man ikke tilknyttes grupper, men har stadig mulighed for at besvare quizzer, og se sit resultat.

På figur 2.1 ses et overblik over grupper, brugere, quizzer og spørgsmål.

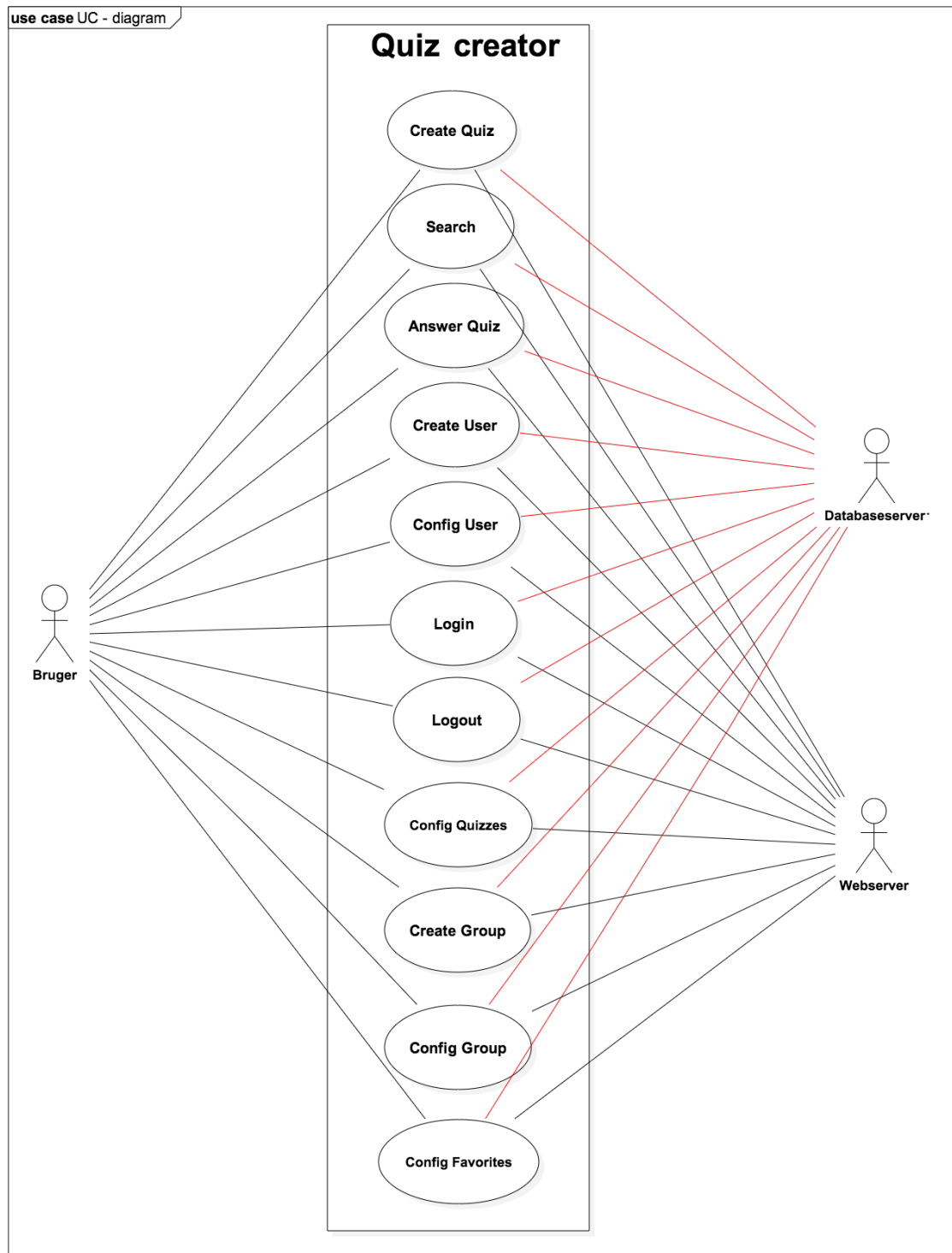


Figur 2.1. Skitsering af produktets logiske opbygning

Her følger de funktionelle og ikke-funktionelle krav til systemet. Kravene er identificeret ud fra valgte brugsscenarier og disses aktører. Hvert scenarie er yderligere beskrevet med use case beskrivelser som specificerer detaljer om forløbene.

3.1 Aktører

Identificerede use cases samt aktører er vist på figur 3.1, hvor primære aktører ses til venstre og sekundære aktører ses til højre. Efter use case-diagrammet findes en beskrivelse over aktørerne, og en gennemgang af hvert use case.



Figur 3.1. Use case diagram med aktører

3.1.1 Aktør-beskrivelse

Her følger en kort beskrivelse af de enkelte aktører vist i use case diagrammet på figur 3.1.

Navn	Bruger
Type	Primær aktør
Beskrivelse	Bruger initierer alle use cases. Det er muligt at oprette en tilbagevendende bruger, eller anvende en midlertidig bruger.

Navn	Databaseserver
Type	Sekundær aktør
Beskrivelse	En server der kan tilgås fra webserveren og som gemmer og håndterer data i databaser.

Navn	Webserver
Type	Sekundær aktør
Beskrivelse	En server der opbevarer og fremsender webapplikationens data på internettet. Den udfører desuden kompileret kode i programmets back end og tilgår databaseserveren.

3.2 Use Case-beskrivelse

Her følger beskrivelse af de enkelte use cases.

UC1: Create quiz	
Mål	Bruger vil oprette en quiz.
Initialisering	Bruger trykker på "Create Quiz" på brugergrænsefladen.
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	Bruger er logget ind
Referencer	UC6: Login
AASH	1 pr. session.
Efterfølgende tilstand	Bruger har oprettet en Quiz.
Hovedforløb	

...fortsat fra forrige side

UC1: Create quiz

1. Webserveren præsenterer de forskellige konfigurationsmuligheder, som Bruger har til rådighed ved "Create Quiz".
2. Bruger indtaster navn på quiz, og dette tilføjes automatisk som et søgbart tag til quizen.
3. Bruger indtaster søgbare tags på quiz.
4. Bruger indtaster spørgsmål.
5. Bruger kan vælge at tilføje et billede til det pågældende spørgsmål ved at trykke på "Gennemse..." og vælge det ønskede billede.
6. Bruger indtaster svarmuligheder, angiver det/de korrekte svar, og tildeler point til de ønskede svarmuligheder.
7. Bruger kan tilføje flere svarmuligheder ved at trykke "Add new answer", hvormed punkt 6 gentages.
8. Bruger kan tilføje flere spørgsmål ved at trykke "Add new question", hvormed der hoppes til punkt 4.
9. Bruger trykker på "Create Quiz" knappen. **Undtagelse 9.a**
10. Webserveren gemmer quizen på databaseserveren og UC afsluttes.

Undtagelser

- 9.a De indtastede oplysninger om quizen er ikke i overensstemmelse med de ikke-funktionelle krav (se afsnit 3.3, side 19).

Gå til UC1 punkt 2

UC2: Search

Mål	Bruger vil finde en ønsket quiz eller gruppe.
Initialisering	Bruger aktiverer søgefeltet på brugergrænsefladen.
Aktører og Stakeholders	Bruger, Databaseserver, Webserver.
Preconditions	
Referencer	
AASH	1 pr. session.
Efterfølgende tilstand	Bruger har markeret den ønskede Quiz.

Hovedforløb

1. Bruger indtaster navnet på de tags, som den ønskede quiz og/eller gruppe er tilknyttet. **Undtagelse 1.a**
2. Bruger trykker på "SØG"knappen på brugergrænsefladen.
3. Webserveren søger efter quizzes og grupper der matcher søgeargumenterne på databaseserveren.
4. Webserveren præsenterer en række quizzes og grupper, som er tilknyttet de tags Bruger indtastede. **Undtagelse 4.a**

...fortsat fra forrige side

UC2: Search

Undtagelser

- 1.a Bruger fjerner fokus fra søgefeltet.
Use casen afsluttes.
- 4.a Søgningen fandt ingen quizzes eller grupper med de angivne tags.
Gå til UC2.1

UC3: Answer Quiz

Mål	Bruger vil besvare en Quiz.
Initialisering	Bruger har markeret en Quiz.
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	Der findes mindst én quiz i databasen
Referencer	UC1: Create Quiz
AASH	1 pr. session.
Efterfølgende tilstand	Bruger har besvaret alle spørgsmål i en Quiz.

Hovedforløb

1. Bruger trykker på navnet på den Quiz han vil besvare.
2. Webserveren henter Quiz fra databaseserveren.
3. Webserveren præsenterer første spørgsmål.
4. Bruger vælger en af svarmulighederne og trykker på "next question". **Undtagelse 4.a**
5. Næste spørgsmål præsenteres for bruger
[Punkt 3 - 5 gentages indtil der ikke er flere spørgsmål i Quizzen]
6. Bruger kan benytte process-prikkerne i toppen af området på brugergrænsefladen til at skifte mellem spørgsmålene.
7. Bruger afslutter Quiz.
8. Webserveren præsenterer statistik der viser hvor mange spørgsmål der er besvaret korrekt og hvor mange point der er samlet ind.
9. Bruger vælger at afslutte hvorved Quiz forlades

Undtagelser

- 4.a Bruger går videre uden at vælge en af svarmulighederne.
Gå til UC3 punkt 5

UC4: Create User	
Mål	Bruger ønsker at oprette en brugerprofil.
Initialisering	Bruger trykker på "Login" og vælger "Register" på brugergrænsefladen.
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	
Referencer	
AASH	1 pr. session.
Efterfølgende tilstand	Bruger bruger oprettet i systemet.
Hovedforløb	
<ol style="list-style-type: none"> 1. Bruger vælger at oprette en brugerprofil. 2. Bruger tilføjer brugernavn. Undtagelse 2.a 3. Bruger kan tilføje fornavn. 4. Bruger kan tilføje efternavn. 5. Bruger tilføjer email. Undtagelse 5.a 6. Bruger tilføjer adgangskode. Undtagelse 6.a 7. Bruger validerer adgangskode. 8. Bruger gennemfører brugeroprettelse. Undtagelse 8 9. Bruger oprettes i databasen og systemet. 	
Undtagelser	
<ol style="list-style-type: none"> 2.a Brugernavnet eksisterer allerede i databasen Bruger notificeres at brugernavn eksisterer. Gå til UC4 punkt 2 5.a Email eksisterer allerede i databasen Bruger notificeres at email eksisterer. Gå til UC4 punkt 5 6.a Adganskode er ikke valid Gå til UC4 punkt 6 8.a Bruger annullerer brugeroprettelse UC afsluttes 	

UC5: Config User	
Mål	Bruger vil ændre sine registrerede brugerdata
Initialisering	Bruger har trykket på "Edit Account"
Aktører og Stakeholders	Bruger, Databaseserver og Webserver
Preconditions	Bruger er logget ind
Referencer	UC6: Login
AASH	1 pr. session
Efterfølgende tilstand	Bruger har ændret de ønskede data

...fortsat fra forrige side

UC5: Config User
Hovedforløb
<ol style="list-style-type: none"> 1. Webserveren henter data for brugeren fra databaseserveren. 2. Webserveren præsenterer data for brugeren. 3. Bruger retter de ønskede data. Undtagelse 3 4. Bruger vælger at gemme data. Undtagelse 4 5. Webserveren gemmer de opdaterede data på databaseserveren.
Undtagelser
<ol style="list-style-type: none"> 3.a Bruger navigerer væk fra konfigurationssiden. UC afsluttes. 4.a Bruger vælger at fortryde ændringerne. Bruger trykker på "Cancel" UC afsluttes.

UC6: Login	
Mål	Bruger ønsker at logge ind.
Initialisering	Bruger trykker på "Login" på brugergrænsefladen.
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	Bruger har oprettet en bruger eller har en Facebook konto
Referencer	UC9: Create User
AASH	1 pr. session.
Efterfølgende tilstand	Bruger er logget ind.
Hovedforløb	
<div>1. Der vises en login boks med mulighed for at skrive emailadresse og adgangskode, oprette en ny bruger, samt muligheden for login via Facebook. Undtagelse 1</div> <div>a) Login med account</div> <div><div>i. Bruger indtaster emailadresse og adgangskode til sin oprettede bruger og trykker på knappen "Sign in" som ikke indeholder et Facebook logo.</div><div>ii. Brugerens data valideres i forhold til data på databaseserveren.Undtagelse 1(a)ii</div><div>iii. Brugerens brugernavn vises i stedet for "login" knappen i menuen.</div><div>iv. Brugerens er logget ind og use casen afsluttes.</div></div> <div>b) Login med Facebook</div> <div><div>i. Brugerens trykker på knappen "Sign in" med et Facebook logo.</div><div>ii. Brugerens brugernavn vises i stedet for "login" knappen i menuen. Undtagelse 1(b)ii</div><div>iii. Brugerens er logget ind og use casen afsluttes.</div></div>	

...fortsat fra forrige side

UC6: Login
Undtagelser 1(a)ii Validering af login data fejlede, og der vises en fejlmeddelelse til brugeren Use casen afsluttes. 1(b)ii Bruger er ikke logget ind på Facebook. Brugeren indtaster sine Facebook login oplysninger i et Facebook-popup vindue. Gå til UC6 punkt 1(b)ii.

UC7: Logout
Mål Bruger ønsker at logge ud. Initialisering Bruger trykker "Logout" på brugergrænsefladen. Aktører og Stakeholders Bruger, Databaseserver og Webserver. Preconditions Bruger er logget ind Referencer UC6: Login AASH 1 pr. session. Efterfølgende tilstand Bruger er ikke længere logget ind.
Hovedforløb
1. Bruger klikker på logout knap. 2. Webserveren logger brugeren ud.
Undtagelser

UC8: Config Quizzes
Mål Bruger vil ændre data for en quiz oprettet af denne bruger Initialisering Bruger har trykket på Config Quizzes Aktører og Stakeholders Bruger, Databaseserver og Webserver Preconditions Bruger er logget ind og bruger har oprettet mindst én quiz Referencer UC1: Create Quiz, UC6: Login AASH 1 pr. session Efterfølgende tilstand Bruger har ændret de ønskede data tilknyttet en quiz
Hovedforløb

...fortsat fra forrige side

UC8: Config Quizzes	
<ol style="list-style-type: none"> 1. Webserveren henter de quizzer der er tilknyttet brugeren fra databaseserveren. 2. Webserveren præsenterer de fundede quizzer for brugeren. 3. Bruger kan trykke på redigeringsikonet eller trykke på sletteikonet. Undtagelse 3 <p>Tryk på redigeringsikonet</p> <ol style="list-style-type: none"> a) Webserveren henter alle data for den valgte quiz på databaseserveren. b) Webserveren præsenterer data for brugeren så de kan redigeres. c) Bruger retter de ønskede data. d) Bruger vælger at gemme quizzen. Undtagelse 3d e) Webserveren gemmer de opdaterede data på databaseserveren. f) Gå til punkt 1. <p>Tryk på sletteikonet</p> <ol style="list-style-type: none"> a) Webserveren beder bruger om en bekræftelse på sletningen. b) Bruger bekræfter sletningen. Undtagelse 3b c) Webserveren sletter quizzen og tilhørende spørgsmål og svar på databaseserveren. d) Gå til punkt 1. 	
<p>Undtagelser</p> <p>3.a Bruger navigerer væk fra konfigurationssiden. UC afsluttes.</p> <p>3d.a Bruger vælger at fortryde ændringerne. Gå til punkt 1.</p> <p>3b.a Bruger annullerer sletningen. Gå til punkt 1.</p>	

UC9: Create Group	
Mål	Bruger vil oprette en Group.
Initialisering	Bruger trykker på "Group" på brugergrænsefladen.
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	Bruger er logget ind
Referencer	UC6: Login
AASH	1 pr. session.
Efterfølgende tilstand	Bruger har oprettet en Group.
Hovedforløb	

...fortsat fra forrige side

UC9: Create Group

1. Brugeren trykker på "Create Group".
2. Webserveren præsenterer de forskellige konfigurationsmuligheder, som Bruger har til rådighed ved "Create Group".
3. Bruger indtaster navn på gruppen, og dette tilføjes automatisk som et søgbart tag til gruppen. **Undtagelse 3.a**
4. Bruger indtaster søgbare tags Ved mere end 1 tag separeres de med komma.
5. Brugeren kan tilføje Quizzer han har oprettet.
6. Bruger trykker på Create knappen.
7. Webserveren præsenterer den nye gruppe.

Undtagelser

- 3.a Det indtastede gruppe navn er allerede taget, indtast et andet.

Gå til UC9 punkt 3

UC10: Config Group

Mål	Bruger kan slette, ændre tilstand på quizzer og slette, tilføje tags i gruppen
Initialisering	Bruger har trykket på "Groups"
Aktører og Stakeholders	Bruger, Databaseserver og Webserver
Preconditions	Bruger er logget ind og har oprettet mindst én gruppe
Referencer	UC6: Login, UC9: Create Group
AASH	1 pr. session
Efterfølgende tilstand	Bruger har ændret eller slettet de ønskede quizzer og tags i en given gruppe

Hovedforløb

1. Bruger trykker på redigeringsikonet for den gruppe der ønskes konfigureret.
2. Bruger for vist gruppens navn, quizzer og tags, samt en liste over gruppens medlemmer. **Undtagelse 2.a**
3. Bruger kan vælge at ændre gruppenavnet.
4. Bruger kan ændre tags i gruppen.
5. Bruger kan slette quizzer i gruppen.
6. Bruger kan vælge at aktivere eller deaktivere en given quiz.

Undtagelser

- 2.a Bruger navigerer væk fra konfigurationssiden.

UC afsluttes.

UC11: Config Favorites	
Mål	Bruger vil fjerne favoritter
Initialisering	Bruger trykker på "Favorites" på brugergrænsefladens menu
Aktører og Stakeholders	Bruger, Databaseserver og Webserver.
Preconditions	Bruger er logget ind
Referencer	UC6: Login
AASH	1 pr. session.
Efterfølgende tilstand	Brugerens favoritter indeholder ikke de fjernede favoritter.
Hovedforløb	
<ol style="list-style-type: none"> 1. Webserveren præsenterer de grupper, som Bruger har markeret som favorit. Undtagelse 1.a 2. Bruger trykker på "Remove favorite" ikonet ved den quiz, der ønskes fjernet. Undtagelse 2.a 3. Bruger trykker på "Quizzes" for at se de grupper som er markeret som favorit. Undtagelse 3.a 4. Bruger trykker på "Remove favorite" ikonet ved den quiz, der ønskes fjernet. Undtagelse 4.a 5. Brugeren er færdig med at fjerne favoritter og use casen afsluttes 	
Undtagelser	
<ol style="list-style-type: none"> 1.a Bruger har ikke tilføjet nogle grupper til favoritter. Og der vises ingen. Gå til UC11 punkt 3. 2.a Bruger trykker på gruppe-navnet Use casen afsluttes og gruppen vises. 3.a Bruger har ikke tilføjet nogen quizzes til favoritter. Og der vises ingen. Use casen afsluttes. 4.a Bruger trykker på quiz-navnet Use casen afsluttes og UC3: Answer Quiz påbegyndes. 	

3.3 Ikke-funktionelle krav

Brugbarhed

1. Programmet skal virke på følgende platforme:
 - a) Mozilla Firefox 38 og nyere (PC og Mac)
2. Tekstlængder er begrænset som følger, i karakterer, begge tal inklusiv:

Quiz: 1 til 75 karakterer

Spørgsmål: 1 til 5000 karakter

Tags: 1 til 75 karakter

Adgangskoder: 6 til 75 karakterer

Brugernavne: 2 til 75 karakter

Svarmulighed: 1 til 75 karakter

3. Det er muligt at tilføje tags til en quiz og gruppe
4. Brugernavne skal være unikke
5. Gruppenavne skal være unikke
6. E-mails for brugere skal være unikke

Ydeevne

7. Programmet skal kunne afvikles af minimum 10 samtidige brugere, som hver har 1 session åben

4

Forundersøgelse

Inden systemet blev designet i detaljer gennemførtes en række forundersøgelser for at bestemme teknologier der kunne være brugerbare til udviklingen af applikationen.

4.1 Grafisk brugerflade-teknologier

Følgende teknologier er vurderet i forbindelse med udviklingen af den grafiske brugergrænseflade.

Navn	Pros	Cons
Microsoft WFP	<ul style="list-style-type: none">• Anvendes i I4GUI	<ul style="list-style-type: none">• Kan kun bruges på Windows-platfomen• Kræver .NET-frameworket på klientens computer
Microsoft Silverlight	<ul style="list-style-type: none">• Stærkt framework til grafisk præsentation	<ul style="list-style-type: none">• Udvikles ikke længere af Microsoft• Kræver Silverlight-frameworket på klientens computer
HTML5 og CSS3	<ul style="list-style-type: none">• Anvendes i I4GUI• Den gængse teknologi for web-applikationer• Understøttet af de fleste platforme	<ul style="list-style-type: none">• Mindre grafiske evner

Ud fra ovenstående er HTML5 og CSS3 valgt for at få en bredt understøttet teknologi som har rigelige muligheder for programmets behov og anvendes generelt mest, af de belyste teknologier.

4.2 Database-teknologier

Følgende teknologier er vurderet i forbindelse med persistering af data.

Navn	Pros	Cons
Microsoft SQL Server	<ul style="list-style-type: none">• Anvendes i I4DAB• Gratis server igennem Microsofts Azure [Microsoft, 2015a]• Integreret med Entity Framework	<ul style="list-style-type: none">• Licenseret brug
MySQL	<ul style="list-style-type: none">• Meget anvendt inden for webudvikling i sammenhæng med PHP• Webbaseret GUI med phpMyAdmin [Müller et al., 2015]• Open-source	<ul style="list-style-type: none">• Kompatibilitet med ASP.NET MVC er begrænset• Integration med Entity Framework begrænset
Oracle Database	<ul style="list-style-type: none">• Integration med Entity Framework	<ul style="list-style-type: none">• Licenseret brug

Ud fra ovenstående er Microsoft SQL Server valgt. Dette er begrundet af muligheden for at få adgang til en server igennem Microsoft Azure [Microsoft, 2015a] samt at ASE stiller en server til rådighed. Da undervisningen i faget I4DAB også anvender denne teknologi er det et oplagt valg.

4.3 Systemarkitektur

Applikationens systemarkitektur blev vurderet tidligt i forløbet. Eftersom det er en GUI applikation gav det god mening at vælge en arkitektur som opdelte view logic fra business logic og data manipulationen. Der var her en række valgmuligheder at benytte, bl.a. MVC, MVVM og MVP. Disse tre arkitekturer ligner på mange punkter hinanden og fælles for dem er at de har et eller flere views som repræsenterer brugerens adgang til systemet og et modellag som har til opgave at manipulere med dataene i systemet. Eftersom vi valgte at benytte ASP.NET teknologien [Microsoft, 2014] var det oplagt at benytte MVC da Microsoft allerede har bygget et af startprojekterne op omkring denne arkitektur. I denne arkitektur er hvert view knyttet til en controller (en controller kan dog godt knyttes til flere views) og ved integration med view-laget sendes requests til den korrekte controller som sørger for at returnere korrekte views og kommunikere med modellaget.

5

Systemarkitektur

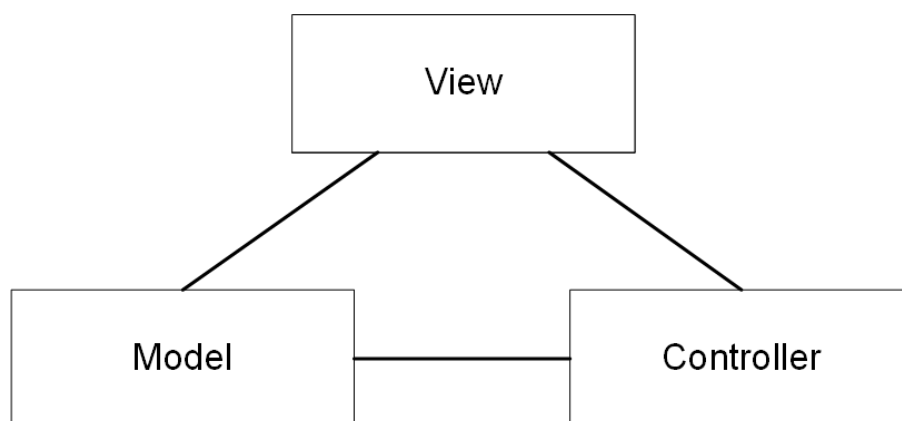
Her følger en detaljeret beskrivelse af programmets arkitektur. Udgangspunktet for beskrivelsen er 4+1 modellen [Kruchten, 1995]. Denne beskriver arkitekturen med fire forskellige abstraktioner med udgangspunkt i use case beskrivelserne og en domænemodel for disse.

5.1 Domænemodel

Domænemodellen fungerer som en visuel ordbog over projektet. De konceptuelle klasser er fundet ud fra aktør-diagrammet og desuden identificeret ud fra use case beskrivelserne. Ud fra domænemodellen kan man analysere sig nærmere ind på systemets problemdomæner. Domænemodellen kan ses i appendiks A på side 75.

5.2 Logical View

Systemet udvikles på ASP.NET MVC-frameworket [Microsoft, 2014]. Her inddeles systemet i tre lag, som vist på figur 5.1.



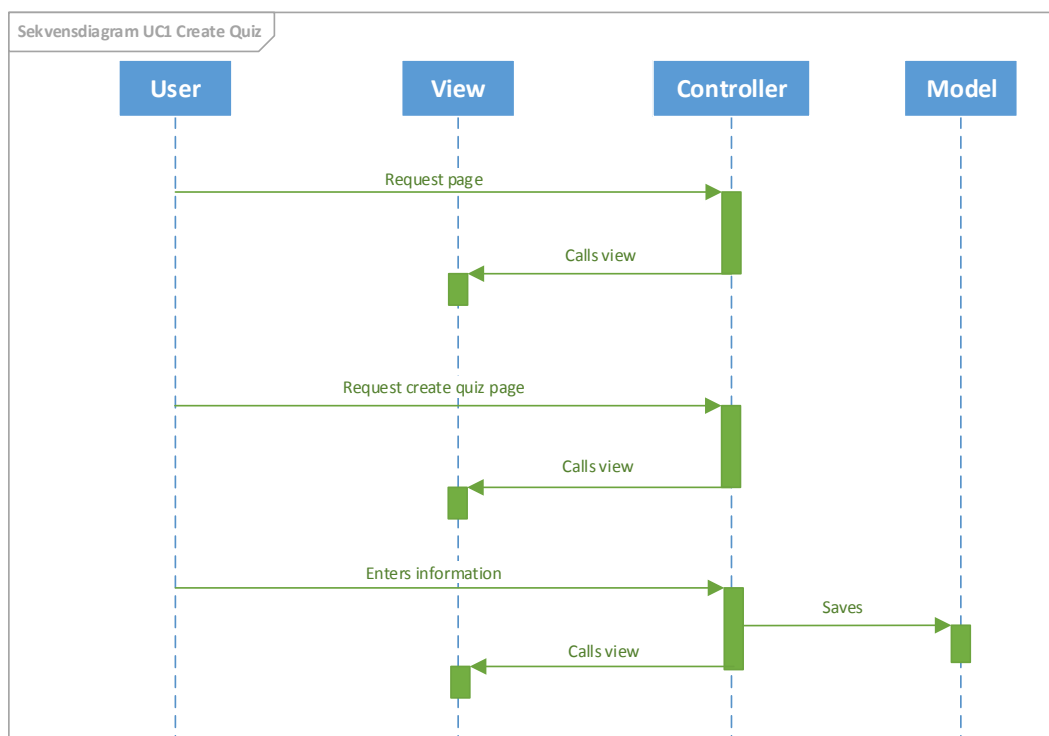
Figur 5.1. Logical View for systemets overordnede arkitektur

Modellaget har ansvaret for adgang til persisteret data. Her anvendes en databaseserver i form af Microsofts SQL Server [Microsoft, 2015d]. I modellaget gemmes også ViewModels, som er View-specifikke modeller. Disse er uafhængige af systemets logik, men bruges udelukkende som bindeled i mellem views og controllers, hvis den oprindelige model ikke er tilstrækkeligt.

View laget håndterer den grafiske fremvisning for brugeren. Her anvendes HTML5 og CSS3 standarderne til formatering af udseendet. I view laget ligger også client side logic i form af JavaScripts. Forbindelsen i mellem View- og Model-laget er udtryk for at viewet validerer model-input så de lever op til de krav der er for de enkelte modellers properties.

Controller laget har ansvaret for business logik og håndterer UC-forløbene. Systemet er udviklet til at have en controller per use case.

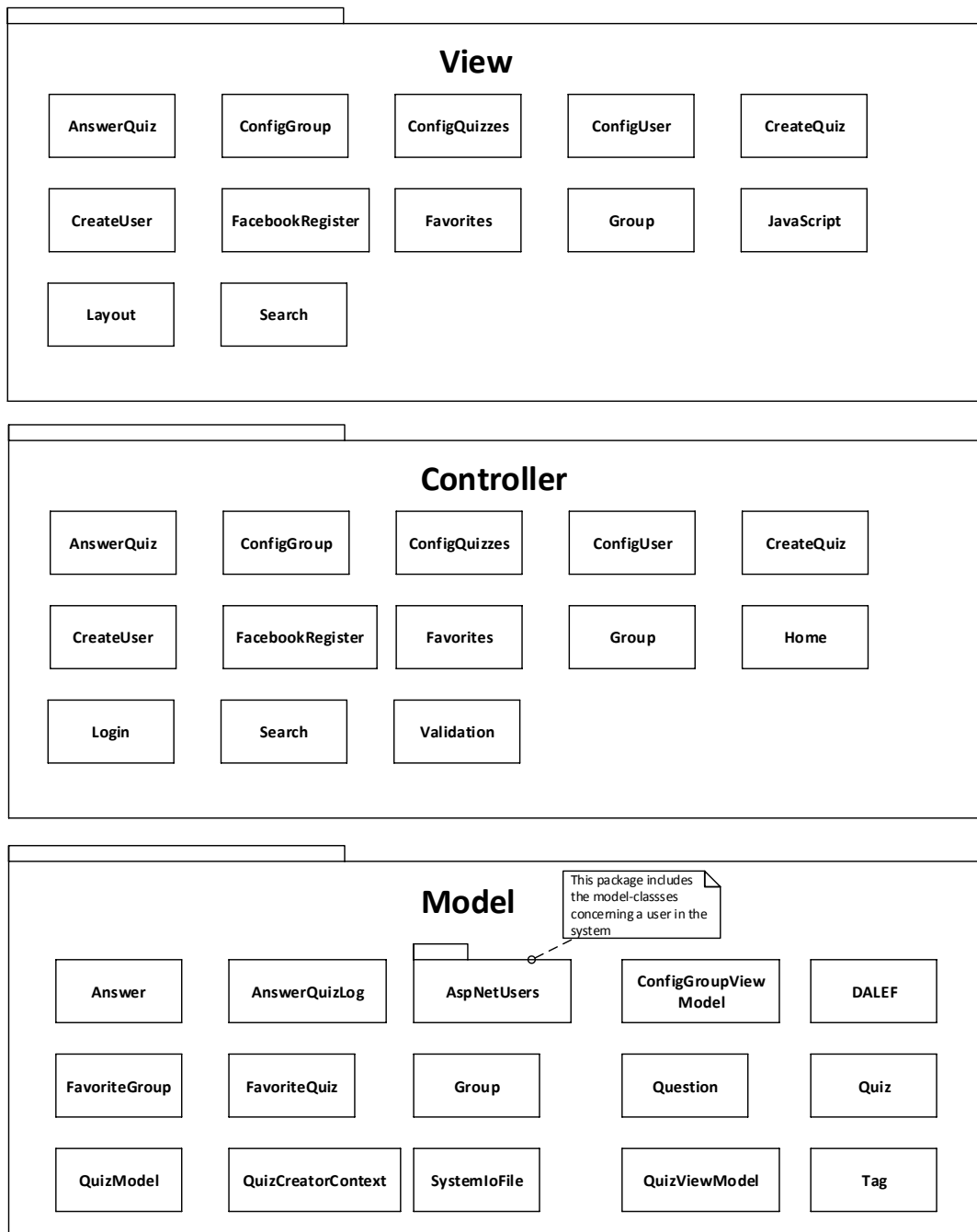
På figur 5.2 er vist et simpelt systemsekvensdiagram for use case 1. Controlleren bliver aktiveret ved et request fra en client ud fra en given URL. Her efter udfører den de nødvendige funktionaliteter og returnerer et givent view eller kalder en anden controller.



Figur 5.2. Systemsekvensdiagram for UC1 Create Quiz

5.3 Implementation View

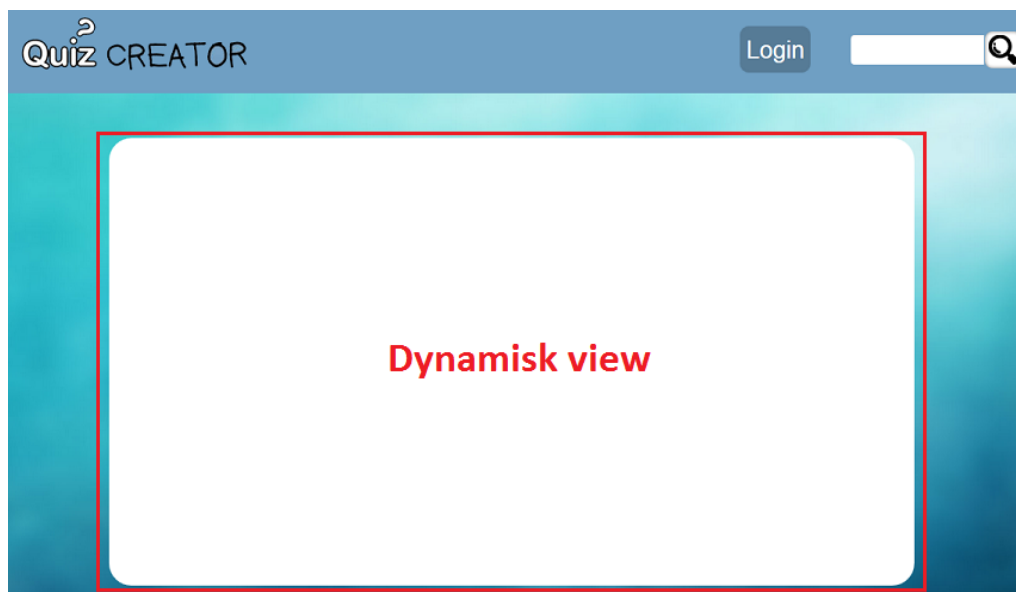
Implementation View er en abstraktion af modulerne i systemet. På figur 5.3 er vist de identificerede pakker i systemet.



Figur 5.3. Implementation View af systemet med identificerede pakker og klasser

View-pakken håndterer web-layoutet i form af HTML og CSS kode. Her styres præsentationens udseende. Pakken indeholder også client side logik i form af JavaScript kode.

Hvert view, undtagen Layout, går ind og sætter indholdet i den dynamiske ramme set på figur 5.4. Viewet Layout er ansvarlig for at sætte den udenomstående grafik. I denne findes logik til at sætte menuen alt afhængig af om brugeren er logget ind eller ej. En detaljeret beskrivelse af de enkelte views er i afsnit 7.3, på side 42.



Figur 5.4. GUI er opdelt i en statisk og en dynamisk del, hvor den statiske del er sat af et layout view

I controller-pakken ligger en klasse for hver use case som håndterer de individuelle forløbshændelser og interaktionen imellem Views og Model. Bemærk dog at use casene for login og logout er slået sammen i én controller.

Model-pakken håndterer modelering af data der skal persisteres i databasen, samt View Modeller, der er modificerede modeller som er specialiserede til et givent view.

5.4 Data View

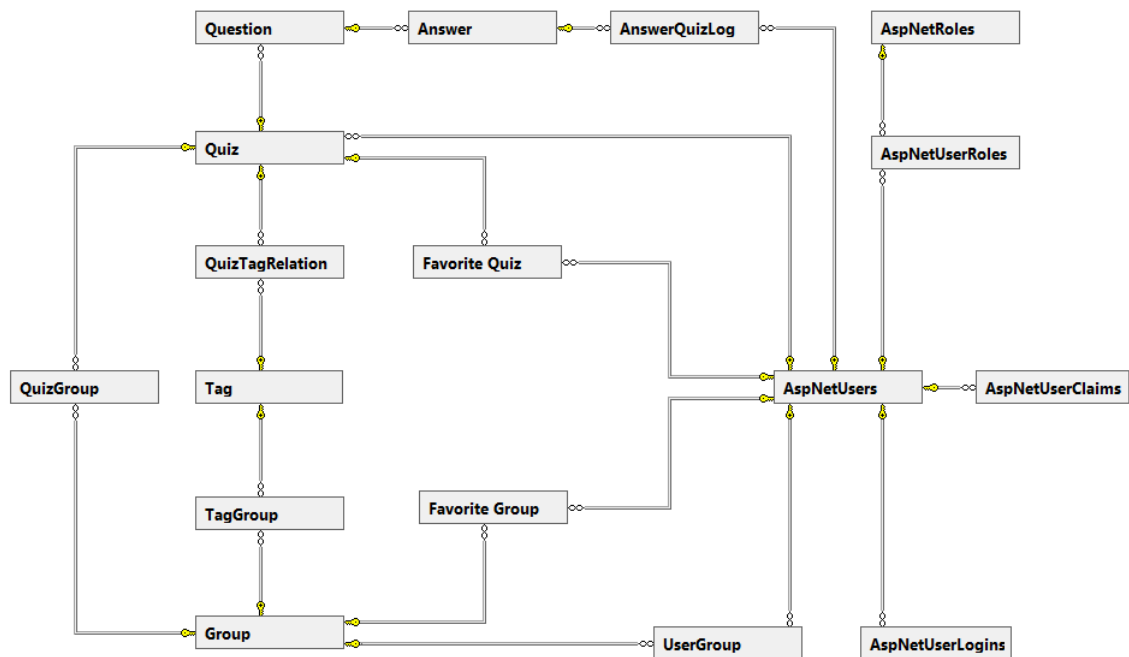
Til systemet bruges persistering af data vha. en database på en databaseserver. De oprettede entiteter/tabeller ses på figur 5.5. Diagrammet benytter relationship-notation i overensstemmelse med Microsoft SQL Management Studio 2014 [Microsoft, 2015d].

Designet er opbygget på baggrund af domænemodellen fra afsnit 5.1, side 23. Entiteterne er identificeret ud fra domæneklasserne fra domænemodellen, og der er oprettet weak entities til entiteter med en mange-til-mange relation. Disse har ingen speciel notation på diagrammet, men består af entiteterne:

- QuizGroup
- QuizTagRelation
- UserGroup
- TagGroup
- AspNetUserRoles

Disse weak entities bruges som en slags opslags-entitet, som er nødvendig ved mange til mange relationer. De holder primary keys for de to entiteter den binder sammen.

I diagrammet på figur 5.5 ses en række entiteter, som alle starter med "AspNet", disse entiteter er skabt i forbindelse med oprettelsen af brugere og login, som benytter .NET Identity 2.0 frameworket [Microsoft, 2015c].



Figur 5.5. Diagram over databasens design, som er brugt til systemet

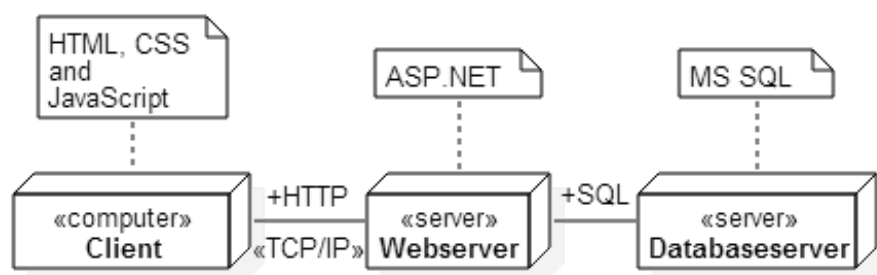
Entiteternes attributter og typer kan ses i Doxygen-dokumentation. Se afsnit 10 på side 73.

5.5 Deployment View

Systemet består af tre fysiske noder, som vist på figur 5.6. På Client-noden udføres den grafiske præsentation af HTML- og CSS-filerne samt client side logic med JavaScript. Filerne tilgås i en webbrowser.

Webserveren genererer HTML- og CSS-filerne til klienten. Den udfører også business logic og håndterer kommunikation med databaseserveren. Databaseserveren håndterer alle persisterede data i systemet. Bemærk at Webserver og Databaseserver ikke nødvendigvis er delt op i to fysiske noder. Det er dog ikke væsentligt for implementeringen hvordan denne opdeling er.

Kommunikationen i mellem Client og Webserver sker over HTTP-protokollen og imellem Web- og Databaseserveren bruges SQL.



Figur 5.6. Deployment View for systemets fysiske noder

Her følger en detaljeret beskrivelse af systemets design. Fra systemarkitekturen og use case beskrivelserne genereres en række sekvensdiagrammer som viser kommunikationen i mellem systemets klasser og lag. Der udvikles ikke komplette applikationsmodeller for de enkelte use cases, da det ikke blev fundet nødvendigt at lave state machines for systemet og at sekvensdiagrammerne var tilstrækkeligt beskrivende. Der er også en lille sammenhæng i mellem klasserne for hver UC hvorfor klassediagrammer for hvert UC også er udeladt.

6.1 QuizModel

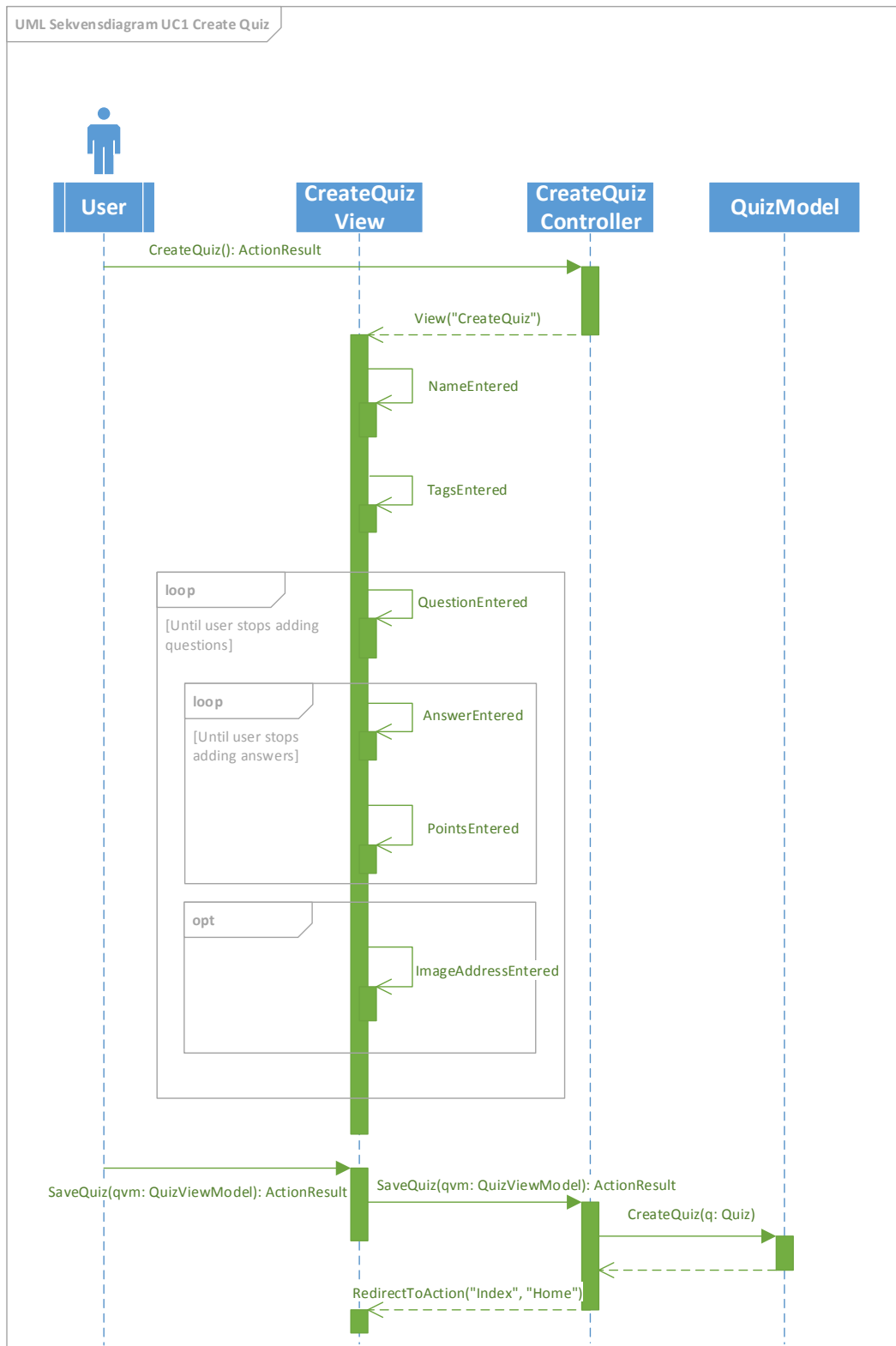
I kravene for applikationen er der identificeret mange entiteter som skal persisteres i databasen. For at give controllerne en simpel adgang til disse funktionaliteter og samtidige abstrahere en del af ansvaret ud fra disse, oprettes en facade til DAL. Dette er en designstrategi beskrevet som Façade Pattern i bogen Design Patterns [Gamma et al., 2005, s. 185-193].

Klassen bryder med Single Responsibility i SOLID-principperne [Wikipedia, 2015], men da facaden betegnes som et interface til sub-modulet DAL er dette en acceptabel løsning.

6.2 Sekvensdiagrammer

CreateQuiz

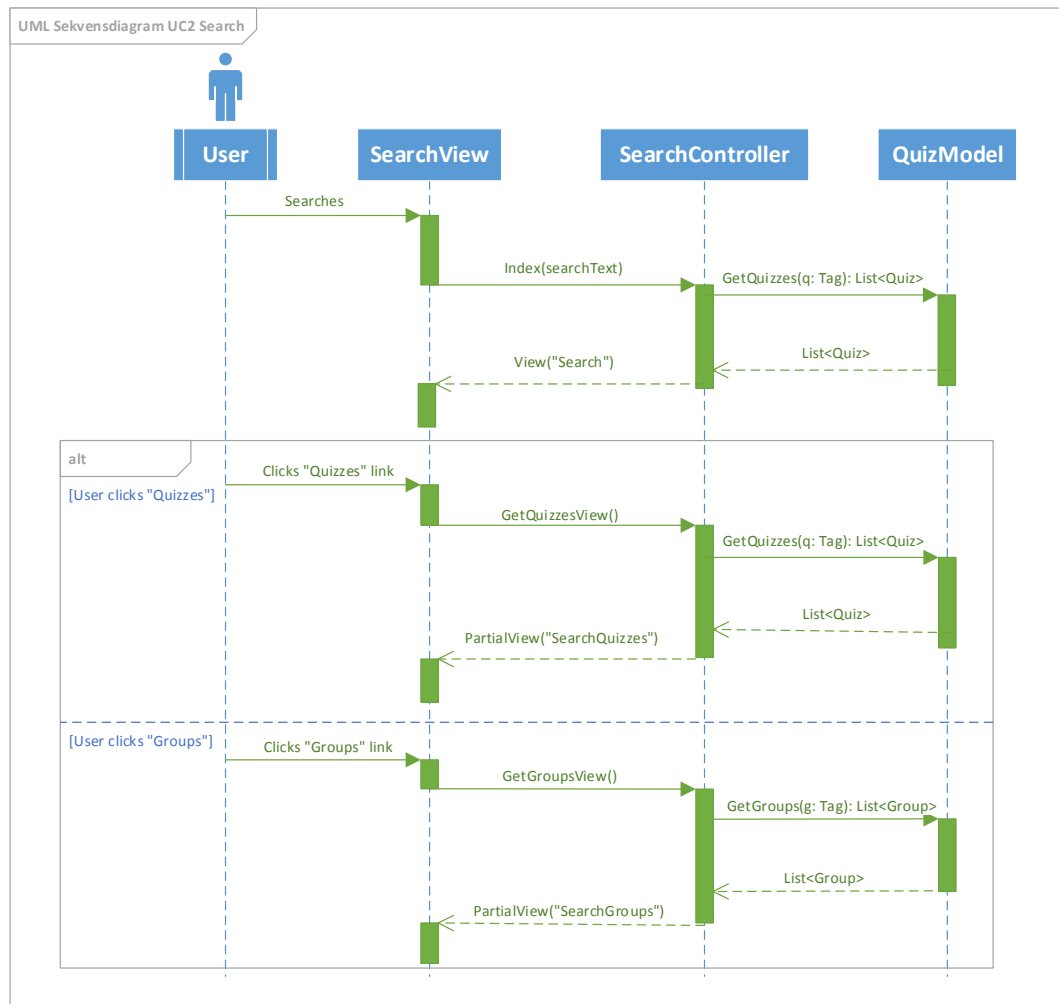
Sekvensdiagrammet vist i figur 6.1 beskriver forløbet for når en bruger ønsker at oprette en quiz i systemet. Som det fremgår af sekvensdiagrammet kan brugeren tilføje flere end et enkelt spørgsmål.



Figur 6.1. Sekvensdiagram for use case 1

Search

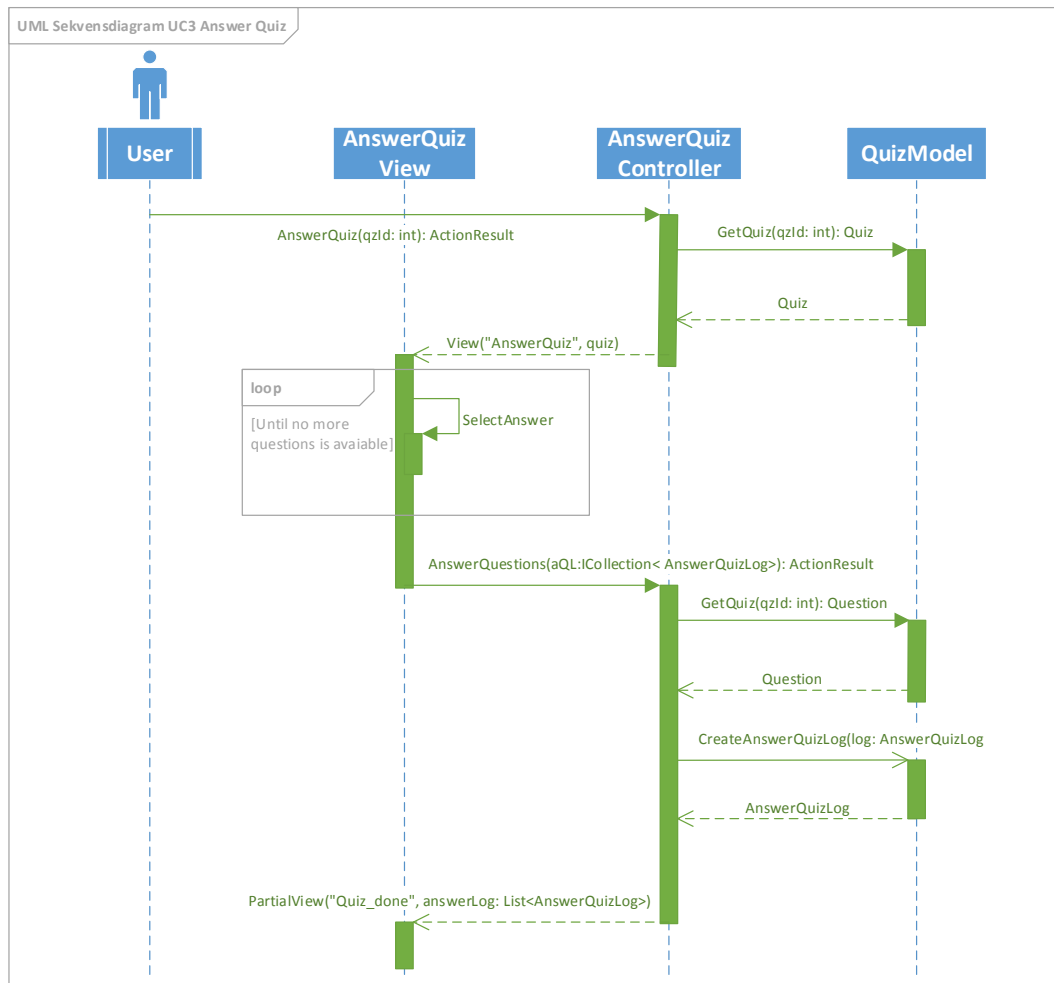
Sekvensdiagrammet vist i figur 6.2 beskriver forløbet for når en bruger ønsker at søge efter en quiz eller gruppe .



Figur 6.2. Sekvensdiagram for use case 2

AnswerQuiz

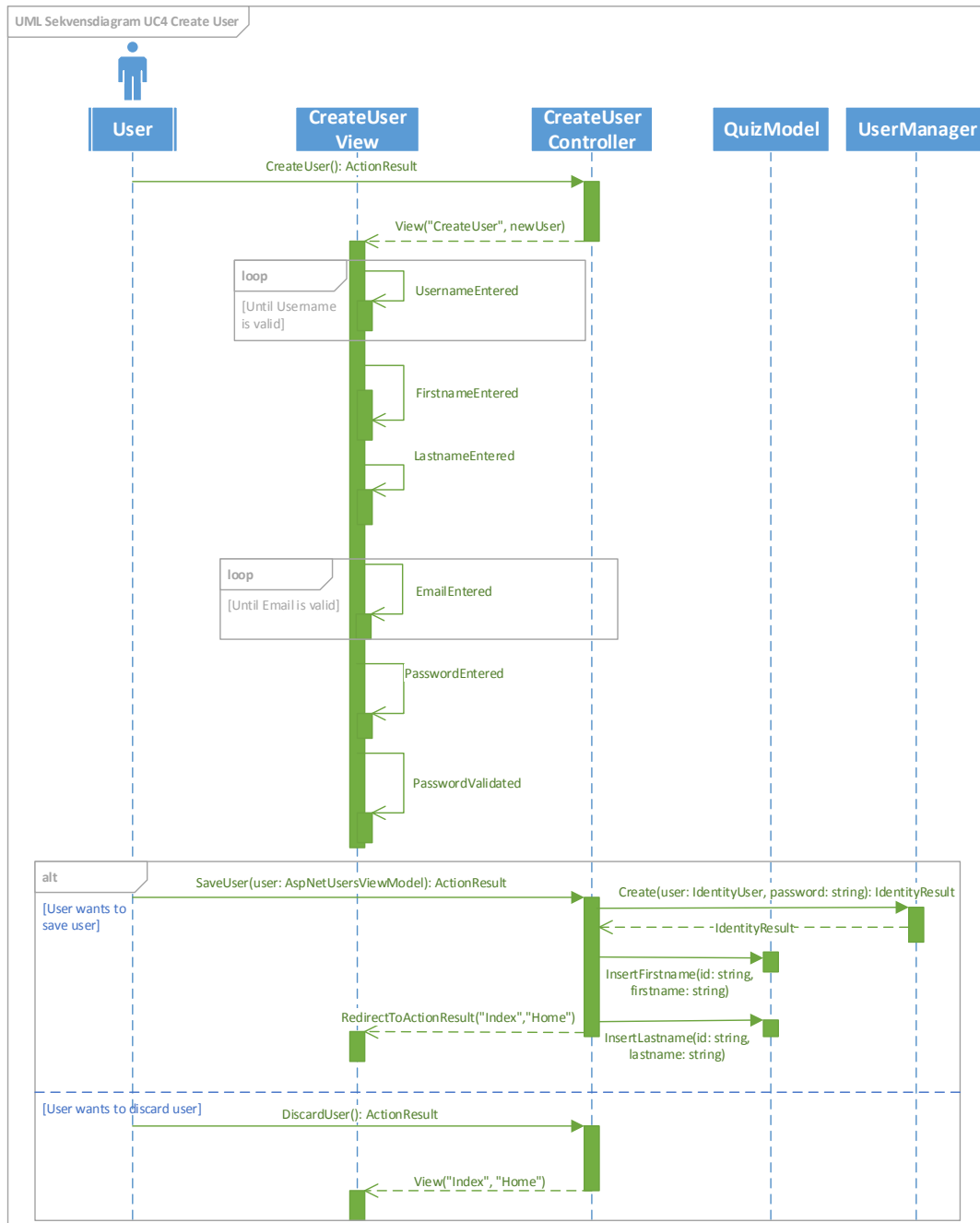
Sekvensdiagrammet vist i figur 6.3 beskriver forløbet for når en bruger ønsker at svare på en quiz. Brugeren vælger den quiz som denne ønsker at svare på hvorefter der svares på det antal spørgsmål quizen indeholder. Efter der er svaret på sidste spørgsmål bliver brugeren præsenteret for det resultat brugeren har opnået igennem quizzzen.



Figur 6.3. Sekvensdiagram for use case 3

CreateUser

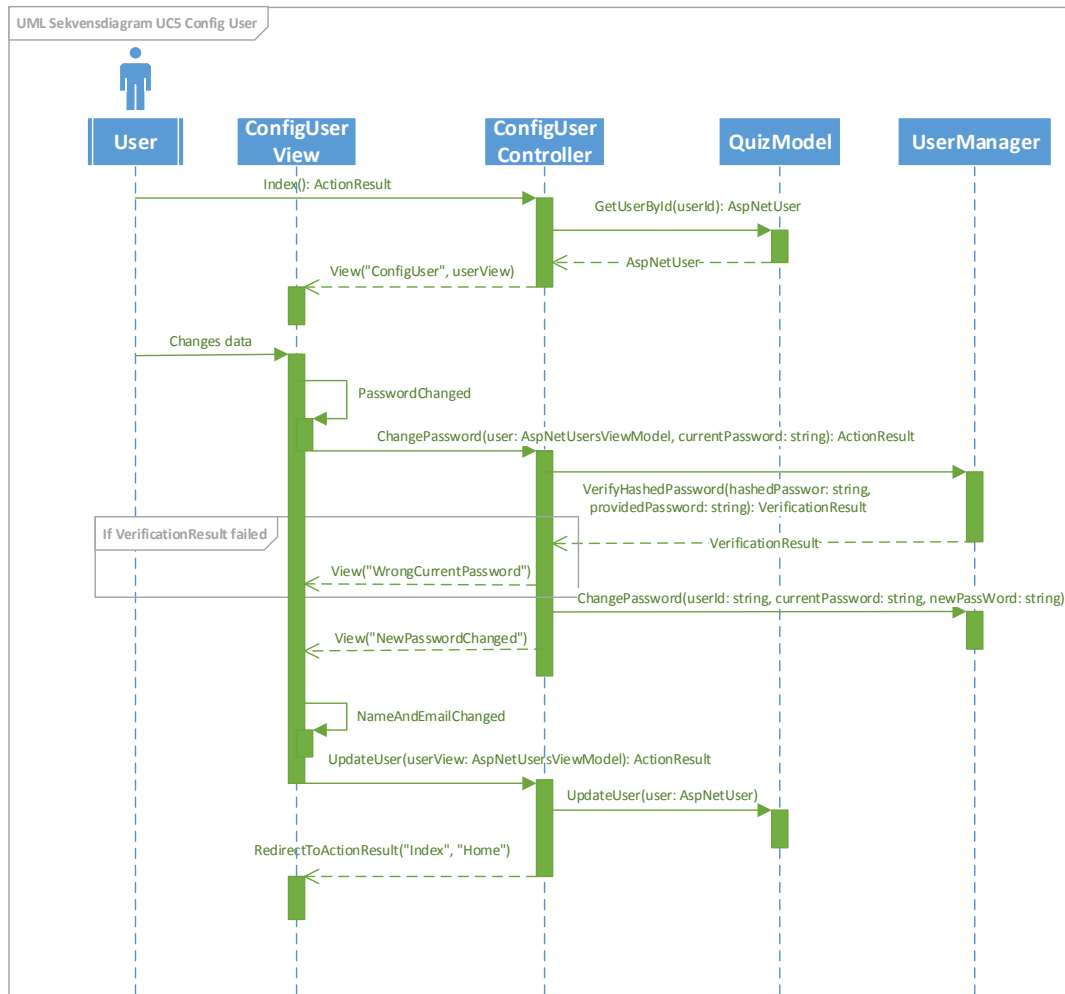
Sekvensdiagrammet vist i figur 6.4 beskriver forløbet for når en bruger ønsker at oprette en brugerprofil i systemet. Brugeren indtaster sine oplysninger, hvorefter denne har mulighed for enten at gemme brugerprofilen i systemet eller at annullere oprettelsen. Til brugeroprettelse anvendes **UserManager**-klassen [Microsoft, 2015e], som er en del af Identity frameworket [Microsoft, 2015c]. **UserManager**-klassen indeholder en række metoder til at oprette og genkende brugere. I dette tilfælde anvendes metoden `Create(user: IdentityUser, password: string): IdentityResult`, som kan bruges til at oprette brugere i et systemet.



Figur 6.4. Sekvensdiagram for use case 4

ConfigUser

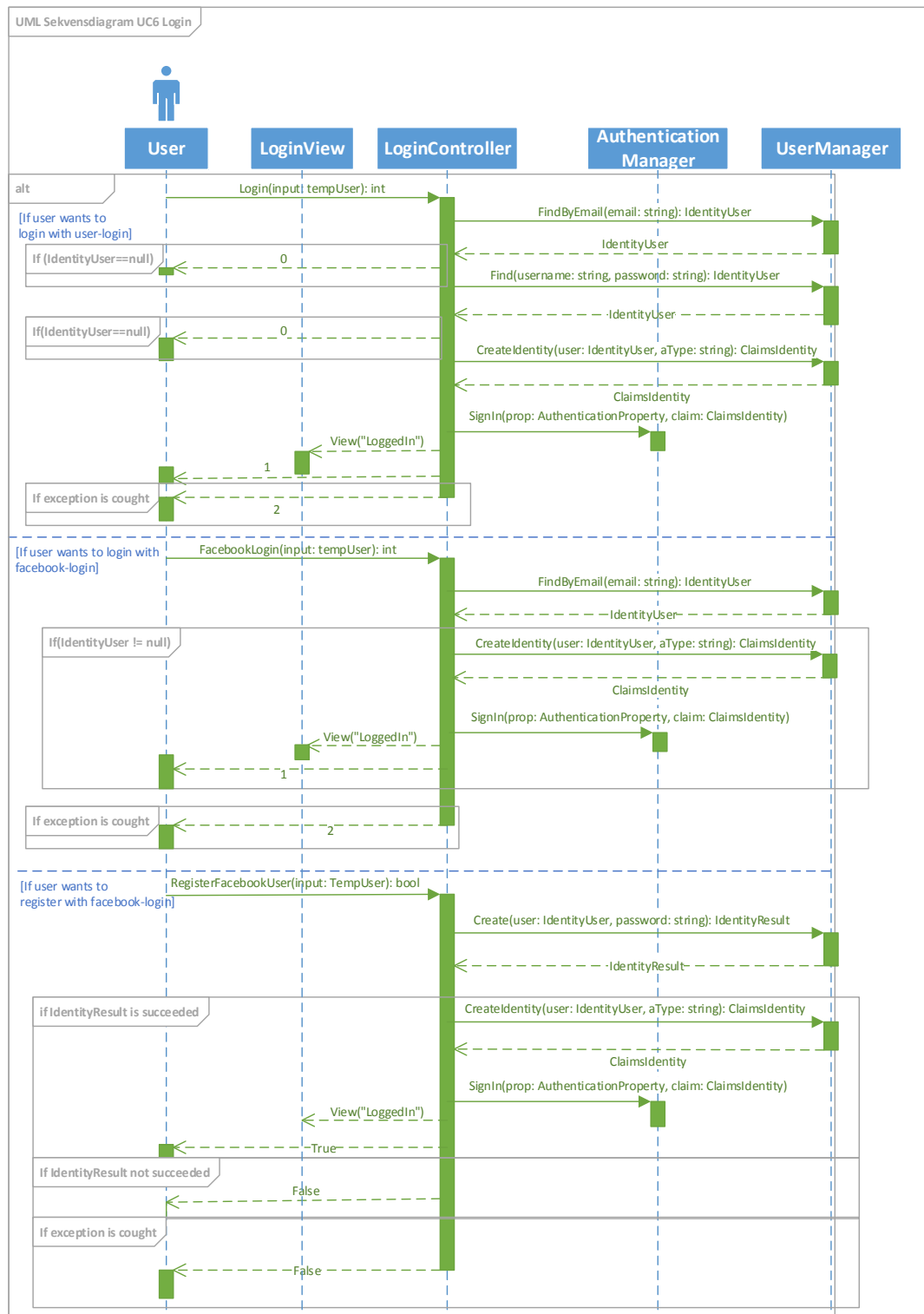
Sekvensdiagrammet vist på figur 6.5 beskriver forløbet for når en bruger ønsker at konfigurere sine oplysninger. Brugeren har mulighed for at ændre sine data og gemme ændringerne.



Figur 6.5. Sekvensdiagram for use case 5

Login

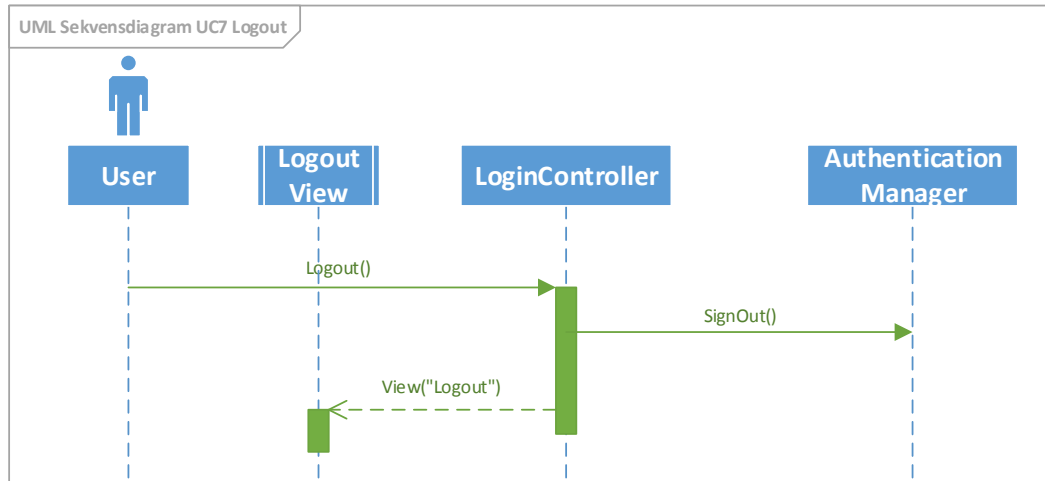
Sekvensdiagrammet vist i figur 6.6 beskriver forløbet for når en bruger ønsker at logge ind i systemet. Her anvendes klasserne **UserManager** og **AuthenticationManager** fra Identity frameworket [Microsoft, 2015b], til genkendelse af brugere der ønsker at logge ind og til godkendelsesprocessen om en bruger eksisterer i systemet.



Figur 6.6. Sekvensdiagram for use case 6

Logout

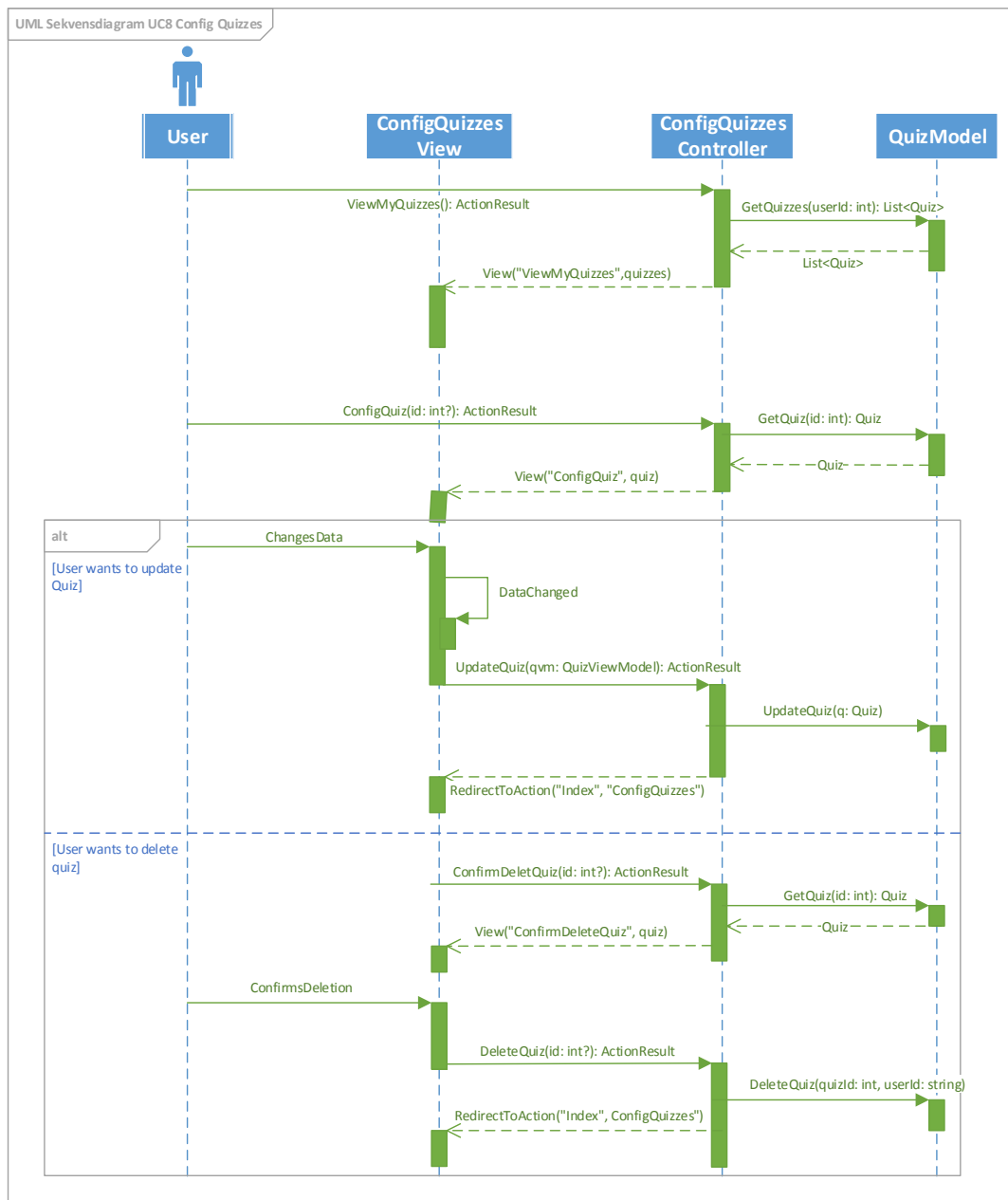
Sekvensdiagrammet vist i figur 6.7 beskriver forløbet for når en bruger ønsker at logge ud af systemet.



Figur 6.7. Sekvensdiagram for use case 7

ConfigQuizzes

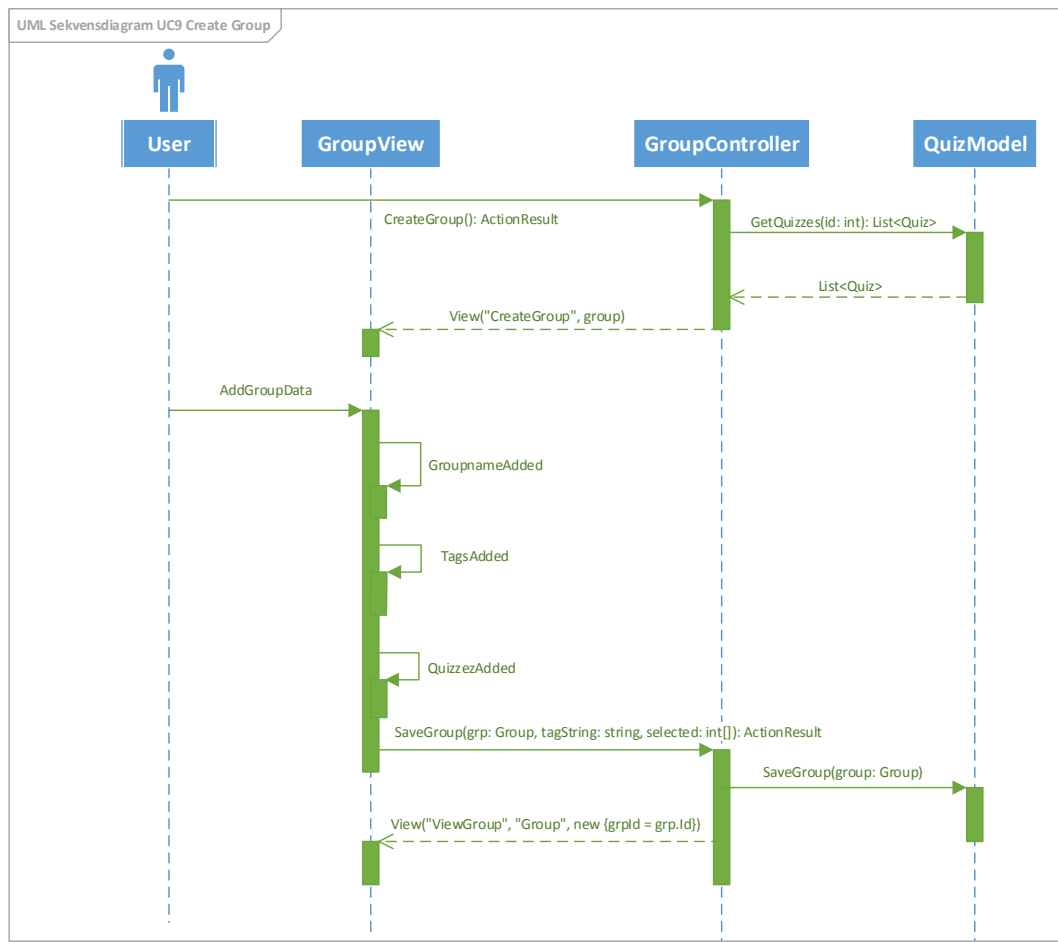
Sekvensdiagrammet i figur 6.8 beskriver forløbet for når en bruger ønsker at konfigurere sine quizzes. Brugeren bliver præsenteret for sine quizzes hvorefter denne har mulighed for enten at ændre på sine quizzes eller at slette en given quiz.



Figur 6.8. Sekvensdiagram for use case 8

CreateGroup

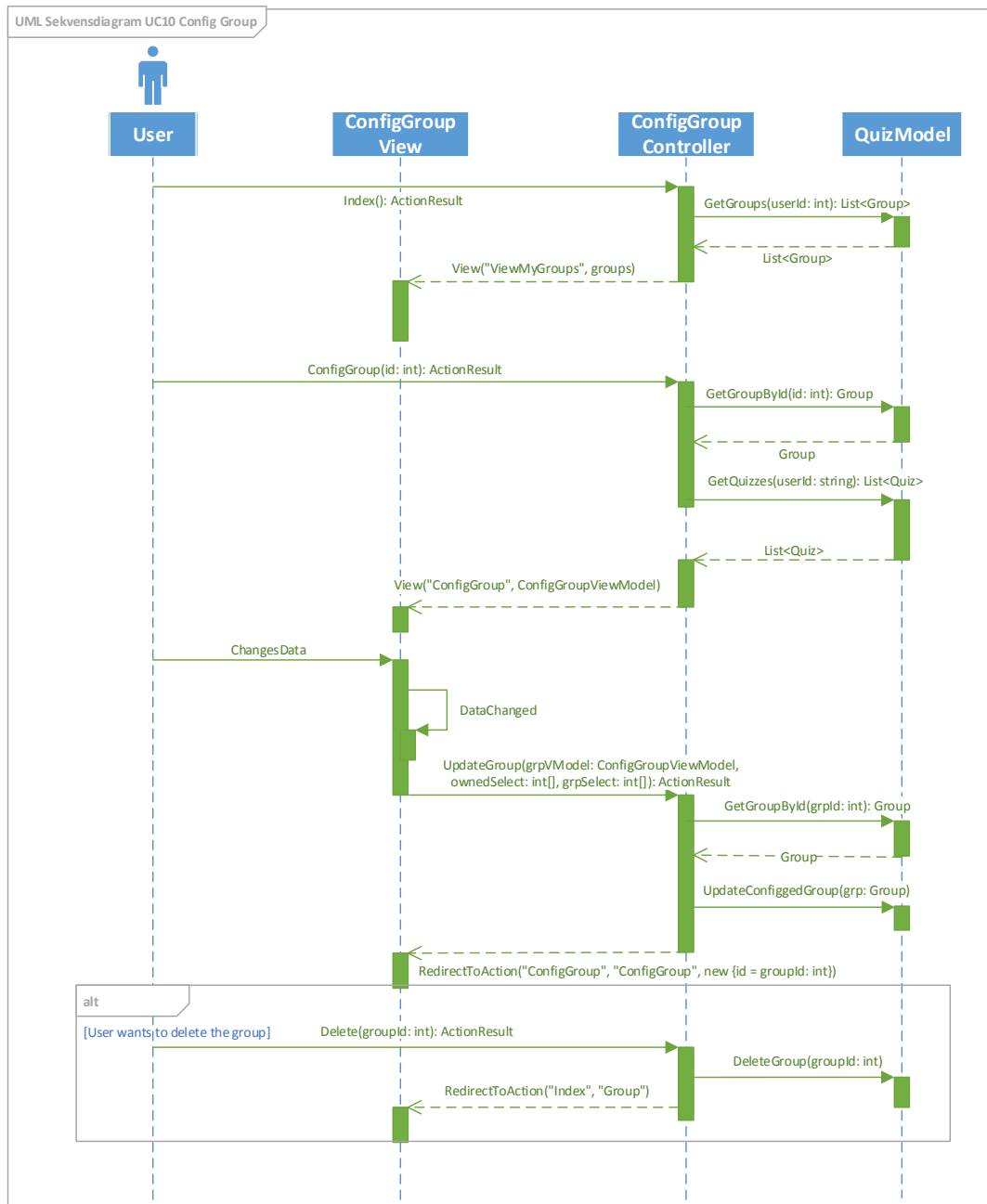
Sekvensdiagrammet i figur 6.9 beskriver forløbet for når en bruger ønsker at oprette en gruppe. Brugeren har mulighed for at tilføje tags, gruppemedlemmer og quizzes til sin gruppe, hvorefter brugeren gemmer sin gruppe i systemet.



Figur 6.9. Sekvensdiagram for use case 9

ConfigGroup

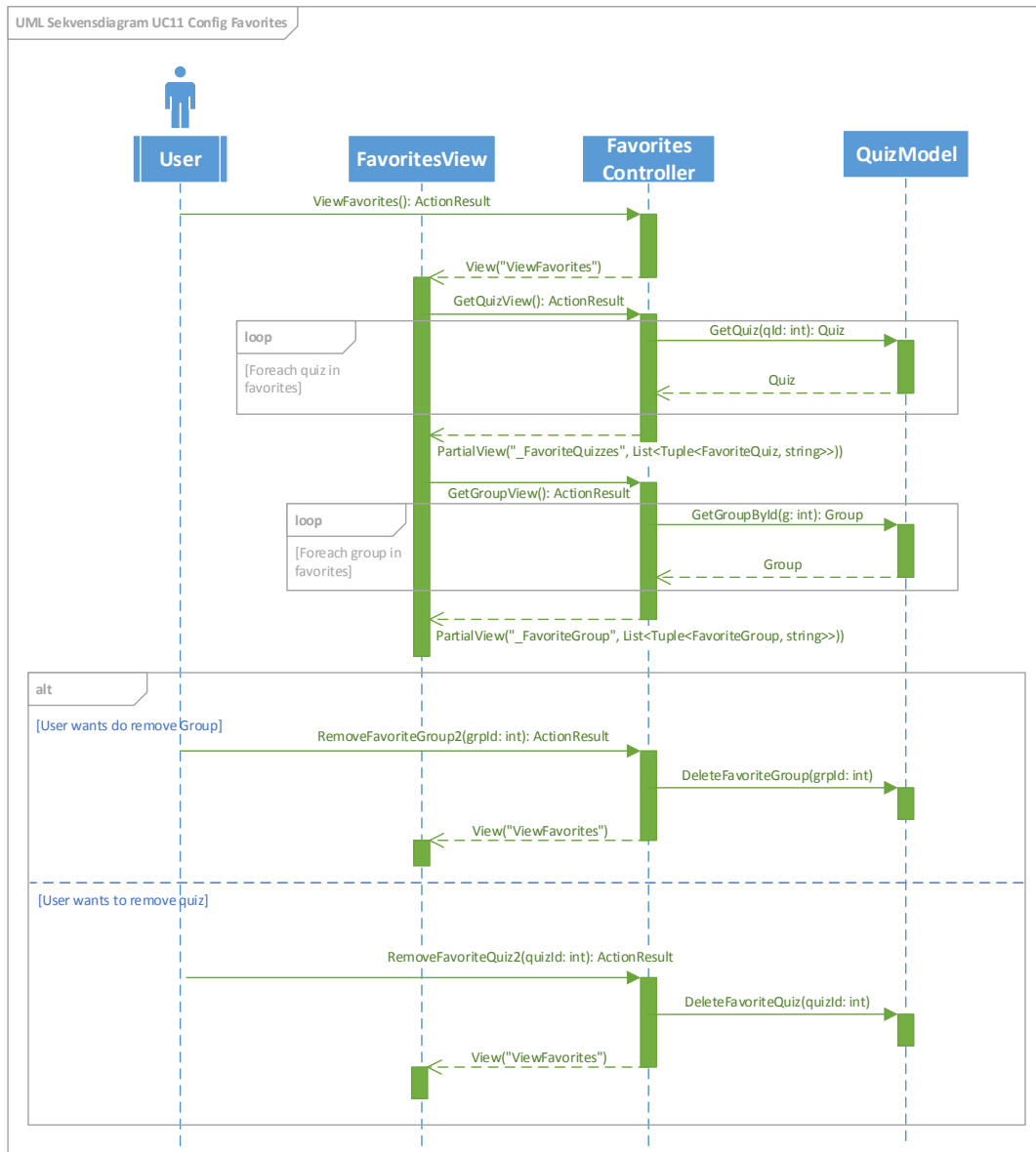
Sekvensdiagrammet i figur 6.10 beskriver forløbet for når en bruger ønsker at konfigurere en af sine grupper. Brugeren bliver præsenteret for sine grupper, og der vælges hvilken gruppe der ønskes at konfigurere. Brugeren har mulighed for ændre på sin gruppe hvorefter brugeren kan vælge at gemme ændringerne.



Figur 6.10. Sekvensdiagram for use case 10

ConfigFavorites

Sekvensdiagrammet i figur 6.11 beskriver forløbet for når en bruger ønsker at slette af sine favoritter. Brugeren bliver præsenteret for sine favoritter hvorefter brugeren kan vælge at slette en eller flere favoritter.



Figur 6.11. Sekvensdiagram for use case 11

7

Implementering

Her følger detaljerede beskrivelser af implementeringen af software.

7.1 Klassediagrammer

Da Klassediagrammerne for systemet er for store og udoverskuelige til at være på et 4-ark, er der lavet en Html-dokumentation til klassediagrammer. Dette kan findes på bilagscd'en i mappen HtmlClassModel og kan åbnes i Firefox. I Html-dokumentationen for klassediagrammer, er der klassediagrammer for Controllers og Modeller både med og uden metoder og attributer. For at se klassediagrammer uden metoder navigeres der til "WebsiteUdenMetoder". Klassediagrammerne med metoder og attributer er at finde i "System" og under de tilhørende namespaces.

7.2 Detaljeret beskrivelse af systemet med Doxygen

Til at dokumentere metoder og klasser, gøres der brug af Doxygen. Ved at tilføje tre skråstreger (///) over klasser og metoder, kan Doxygen generere en HTML side hvori beskrivelser og relationer er vist. På Kodesnippet 7.1 ses et eksempel på beskrivelse af en metode.

Kodesnippet 7.1. Eksempel på Doxygen notation

```
1  /// <summary>
2  /// Get quiz by Id from QuizModel, aggregate tags to string and return
   ConfigQuiz view
3  /// </summary>
4  /// <param name="id">Quiz ID</param>
5  /// <returns>Editor view for the quiz. Redirects to Index() if Id is null
   or non-positive</returns>
6  public ActionResult ConfigQuiz(int? id)
7  {
8      if (!User.Identity.IsAuthenticated)
9          return RedirectToAction("Index", "Home");
10     ...
11 }
```

På Cden følger der en HTML side under Doxygen, hvor dokumentationen præsenteres. På 7.1, ses et screenshot af layoutet.



Figur 7.1. Visning af Doxygen layout

7.3 Beskrivelse af views

Her følger beskrivelser af de views der præsenteres på websiden.

7.3.1 Quiz views

Create Quiz

I Create Quiz viewet kan der oprettes en ny quiz. Viewet er delt op således at Question og Answer er to partialviews, som er fælles for Create Quiz og Config Quiz. Viewet kan ses i figur 7.2. Her udgør Create Quiz viewet alt inden for den røde kasse. Herinde vises Question partial viewet inde i den gule kasse, mens Answer partial viewet ses inde i den grønne kasse.

Create Quiz

Name
I3HAL_Eksamen1

Tags
HAL IKT E2014 eksamen iha igeniorhøjskolen

Create Quiz

1 2 3 4 +

Question: 2

Hvilken "clock mode" og "bits per word" anvender den viste SPI enhed?

Remove question

Attach image
Vælg fil HAL2.PNG

#	Answer	Correct	Points	Remove
1	Clock mode 00b, 8 bits per word	<input checked="" type="checkbox"/>	10	×
2	Clock mode 01b, 4 bits per word	<input type="checkbox"/>	-5	×
3	Clock mode 11b, 16 bits per word	<input type="checkbox"/>	-5	×
4	Clock mode 00b, 12 bits per word	<input type="checkbox"/>	-5	×

Add new answer

Figur 7.2. GUI til create quiz viewet med kasser, der opdeler viewet i partial views. Gul: Question partial view. Grøn: Answer partial view

I Question partial viewet bliver der brugt Bootstrap paging, det fungerer ved at alle divs bliver listet ud efter hinanden, men Bootstrap sørger for kun at vise det der har css-klassen "active" på sig.

Dynamisk generering af inputfelter i forme

Formene i Create og Config quiz er udviklet til at kunne udvides dynamisk ved brug af AJAX og jQuery. I controllerne CreateQuiz og ConfigQuiz, er der metoder som returnerer den nødvendige HTML opbygning. Denne hentes med AJAX og indsættes på siden uden at hele siden behøver at loades igen.

For at udnytte ASP.NETs evne til at arbejde på modeller redigeres navnene på alle inputfelterne så de svarer til de navne der ville være hvis objekterne var oprettet i C#. For eksempel, hvis en quiz har ét spørgsmål og der tilføjes et nyt, får det nye input-felt, til teksten, navnet `Questions[1].Text`. På denne måde kan den samlede håndtering af form-inputtet bruge ét argument af typen `Quiz`, i stedet for at skulle tage alle parametre

ind én af gangen. Et udklip af koden er vist i Kodesnippet 7.2.

På linje 2-5 er AJAX-kaldet. Hvis dette går godt udføres funktionen i linje 5. Her tilføjes HTML-koden fra controlleren en `div`-container. Model-binding navnet konstrueres og tilføjes alle nye `input`-tags. Til sidst returneres `false`, for at undgå at formens almindelige `submit-action` udføres.

Kodesnippet 7.2. JavaScript udklip til AJAX og model-binding håndtering ved indsættelse af nye spørgsmål

```
1 $(document).on("click", "#addQuizQuestion", function() {
2     $.ajax({
3         url: this.href,
4         cache: false,
5         success: function(html) {
6             // Append new question form
7             $("#ConfigQuizzesQuestionsContainer").append(html);
8
9             /* Removed code for simplicity */
10
11             // Create prepend string to fix object binding
12             var prepend = "Questions[" + (newCounter - 1) + "].";
13
14             // Get all input elements and prepend input names
15             $(".ConfigQuizzesQuestion:last input").each(function() {
16                 var original = $(this).attr("name");
17                 $(this).attr("name", prepend + original);
18             });
19
20             /* Removed code for simplicity */
21         }
22     });
23     return false;
24 });
```

Sletning af spørgsmål og svar

Udover dynamisk generering af spørgsmål og svar, kan de også slettes igen. Dette gøres med jQuery, hvor hele klientens HTML bliver manipuleret, og på den måde kan der vælges hvilke divs der skal være "hidden". Derudover skal der rettes i indekseringen af spørgsmål og svar, således at hvis man sletter spørgsmål 1, skal spørgsmål 2 nu blive til nummer 1 osv.

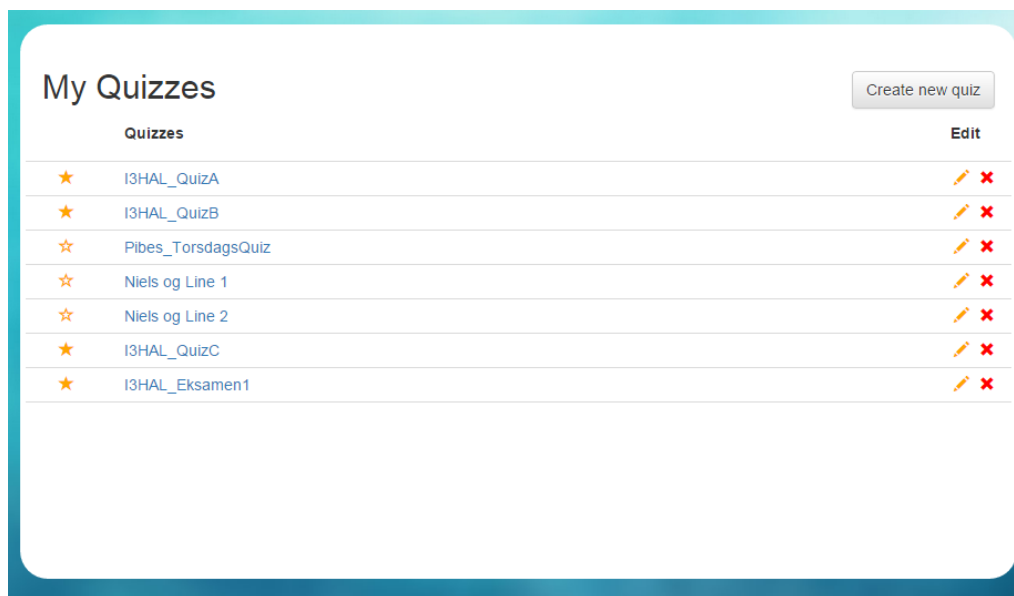
Config Quizzes

Config Quizzes bruger de to samme partial views (Question og Answer) som Create Quiz viewet fra figur 7.2, så de to views er næsten identiske. Config Quiz giver mulighed for at rette i en quiz som den samme bruger selv har oprettet.

Hvis et spørgsmål skal slettes, bliver der med jQuery sat "true" i et "hidden-input", og spørgsmålet bliver gemt. På denne måde kan der i controlleren tjekkes på om spørgsmålet har et "delete flag", og dermed slette spørgsmålet fra databasen. Den samme fremgangsmåde er implementeret for svar og billeder.

View My Quizzes

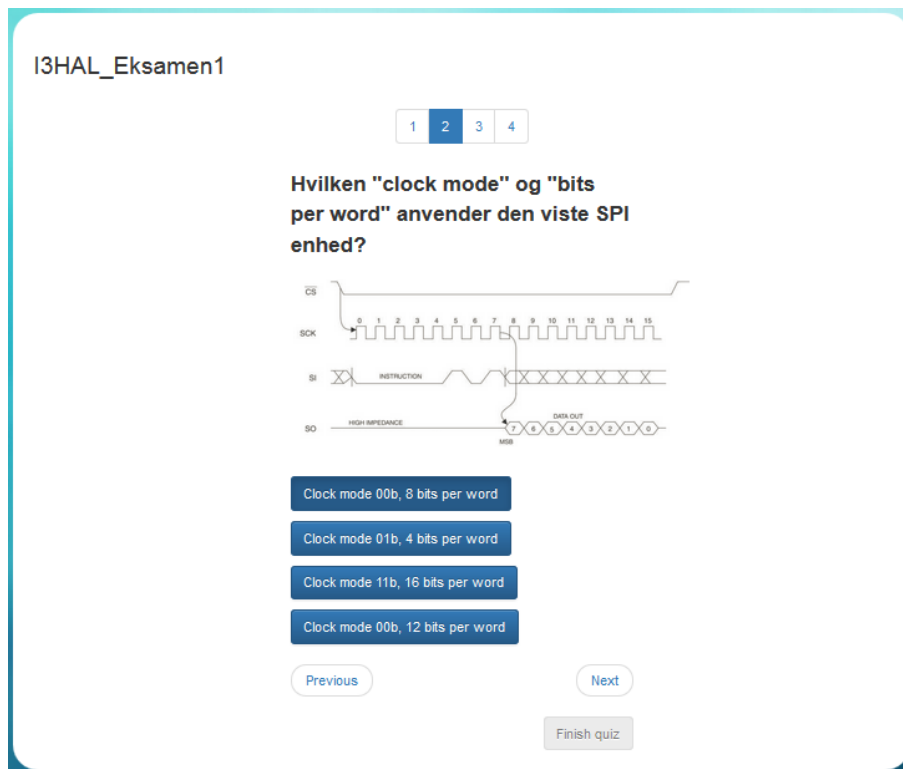
Viewet åbnes ved at trykke på "Quizzes" i menuen som ses på figur 7.3. Det kræver at brugeren er logget ind for at åbne viewet. Hvis det skulle ske, at en bruger der ikke er logget ind forsøger at åbne viewet bliver han returneret til websidens index. Dette view danner en bro mellem brugeren og de quizzer som brugeren selv har oprettet. Viewet tillader brugeren at slette eller ændre sine quizzer. Ved tryk på blyanten under Edit åbnes Config Quizzes viewet for denne quiz. Quiznavnet agerer som link til Answer Quiz viewet med denne quiz.



Figur 7.3. GUI af my quizzes viewet

Answer Quiz

I Answer Quiz, kan der markeres forskellige svar muligheder inden for hvert spørgsmål. Spørgsmålene bliver præsenteret med Bootstrap paging (se figur 7.4, således at der er en list af knapper hvor det ønskede spørgsmål i rækken kan vælges. Derudover kan der også bladres igennem spørgsmålene med en "Previous" og "Next" knap. Knapperne er tilkoblet en jQuery funktion der sætter en "active" klasse på det hhv. forrige og næste spørgsmåls div.



Figur 7.4. GUI ved besvarelsen af en quiz

Brugeren afslutter quizbesvarelsen ved at trykke på knappen "Finish Quiz", som først bliver tilgængelig, når der er markeret en svarmulighed ved alle spørgsmål. Efter tryk på denne knap tages brugeren videre til resultatsiden, som ses i figur 7.5. Her vises en opsummering af hvordan brugeren har klaret sig. Dette gøres ved at vise hvilke spørgsmål der er svaret rigtigt og forkert ved, samt antallet af point der er givet.

Finished I3HAL_Eksamen1			
Question	Your answer	Correct	Points
Hvordan skrives følgende på funktio	char*(*myReader)(int file, int leng	✓	10
Hvilken "clock mode" og "bits pe	Clock mode 00b, 8 bits per word	✓	10
I hvilken folder finder man hardwar	/dev	✓	10
En device driver til en 8-kanal A/D	Ved at benytte et IOCTL kald	✗	0
Total points: 30			

Figur 7.5. GUI ved resultatsiden efter besvarelsen af en quiz

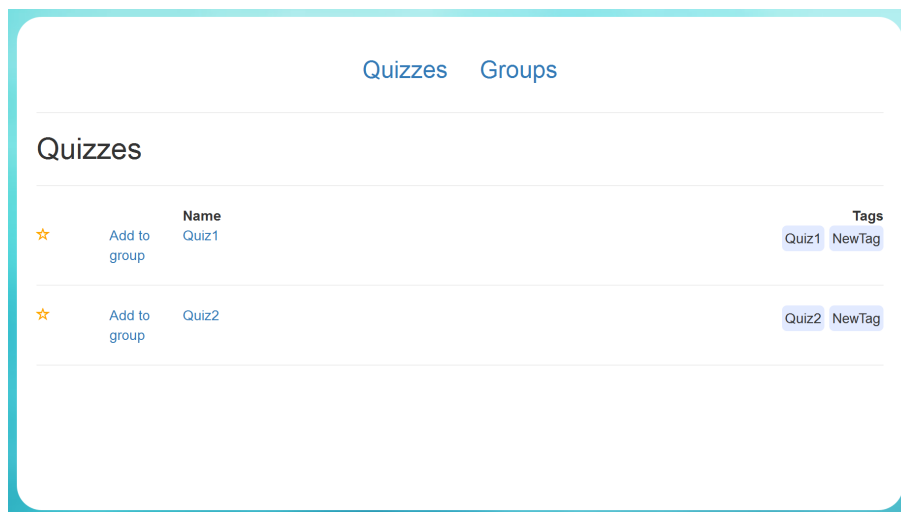
Search

På menubjælken er der et inputfelt hvor bruger kan søge efter quizzer og/eller grupper ved at indtaste ønsket søgeargument. Når bruger trykker på "SØG"knappen bliver bruger præsenteret for Search viewet. Search viewet viser en række quizzer og/eller grupper som matcher brugers søgeargument. Hvis ingen quiz eller gruppe blev fundet vil Search viewet informere bruger om det.

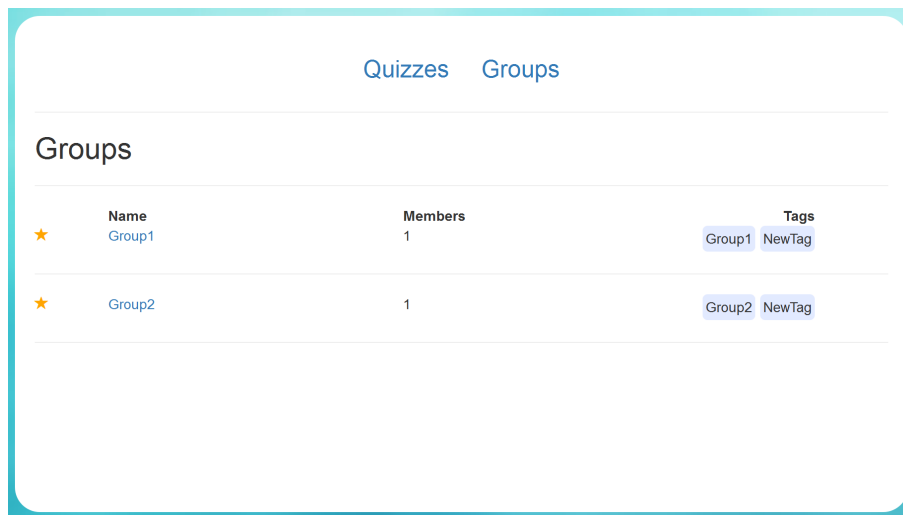
Søgesystemet søger efter quizzer og grupper som har tags der matcher søgeargumentet - quiznavn og gruppenavn indgår også som tag. Søgesystemet har en meget bred søgning. Dvs. at så længe søgeargumentet indgår i en eller flere quizzer og/eller grupper tags, vil de pågældende quizzer og/eller grupper blive præsenteret på Search viewet. Fx hvis søgeargumentet er 'd', vil søgesystemet finde samtlige quizzer og/eller grupper hvor 'd' indgår.

Search viewet består i princippet af to views. Et som viser alle de quizzer som matcher søgeargumentet og det andet viser grupperne. Til visning af de to views bruges bl.a. jQuery AJAX teknologien. Ved brug af jQuery AJAX kan der skiftes imellem de to views uden at hele web applikationen bliver opdateret, fordi client side sender og modtager data til og fra server side asynkron, dvs. at der opdateres kun på det ændrede data og ikke hele web applikationen.

Figur 7.6 viser viewet for Quizzes, og figur 7.7 viser viewet for Groups.



Figur 7.6. Search view - viser Quizzer



Figur 7.7. Search view - viser Groups

7.3.2 Group views

Create Group

For at oprette en ny gruppe bruges Create Group viewet. Det findes ved at trykke på "Groups" og herefter trykke på knappen "Create new group" i øvre højre hjørne på figur 7.10. Det er nødvendigt at være logget ind for at kunne se knappen "Groups".

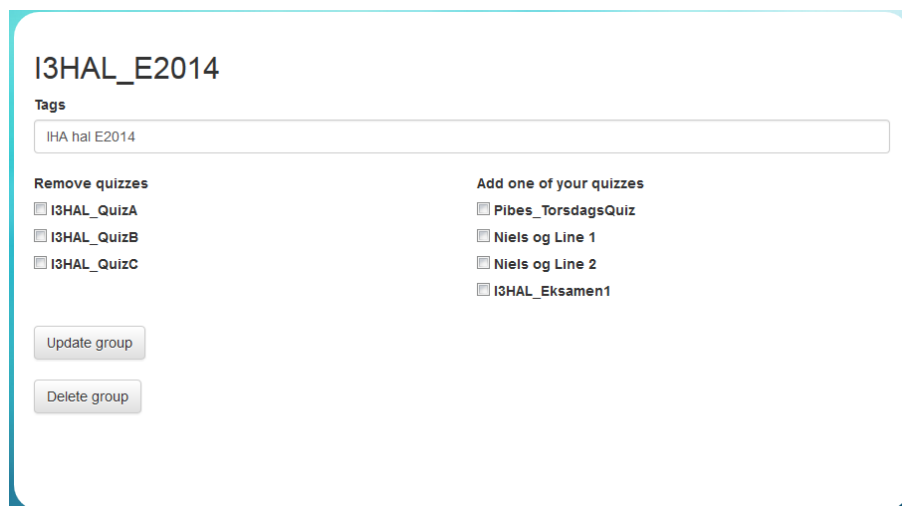
Create Group viewet har en simpel inputform til at oprette en ny gruppe med navn og tags som vist på figur 7.8. Derudover er det muligt at tilføje brugerens egne quizzes med det samme fra listen ud for "Add quizzes".

Der er implementeret en validering af gruppenavnet, som sikrer at brugeren ikke opretter en gruppe med et navn der allerede er registreret. Til dette anvendes ASP.NET-frameworkets Remote Validation funktion, se afsnit 7.4 på side 53.

Figur 7.8. GUI til create user viewet

Config Group

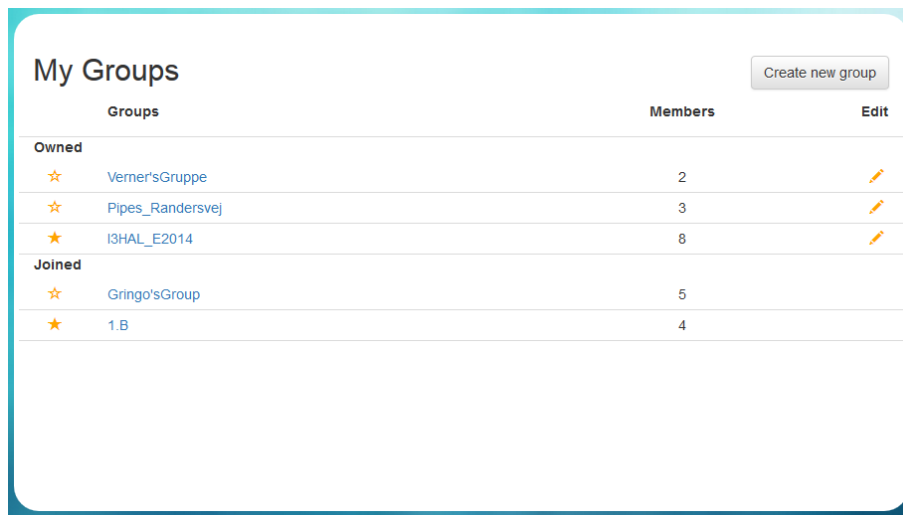
Dette view giver brugeren mulighed for at ændre i informationerne for en gruppe. Billedet er næsten identisk med "Create Group" viewet. Det er muligt at ændre de tilknyttede tags og det er muligt at fjerne quizzer og tilføjer egne quizzer ud fra to lister. Det er ikke muligt at ændre gruppens navn, da dette er besluttet at være bindende når gruppen oprettes. Hvis det ønskes kan gruppen også slettes fra dette view. Se figur 7.9.



Figur 7.9. GUI til config group viewet

View My Groups

Viewet åbnes ved at trykke på "Groups" i menuen som ses på figur 7.10. Det kræver at brugeren er logget ind for at åbne viewet. Hvis det skulle ske, at en bruger, der ikke er logget ind forsøger at åbne viewet bliver denne returneret til websidens index. Viewet tillader brugeren at benytte de grupper, som brugeren selv har oprettet samt de grupper, som brugeren har tilmeldt sig. Viewet tillader brugeren at ændre sine grupper ved at trykke på blyanten under Edit, hvor Config Group viewet for denne gruppe åbnes. Dette er kun muligt for de grupper som brugeren selv har oprettet. Gruppenavnet agerer som link til View Group viewet med denne gruppe.

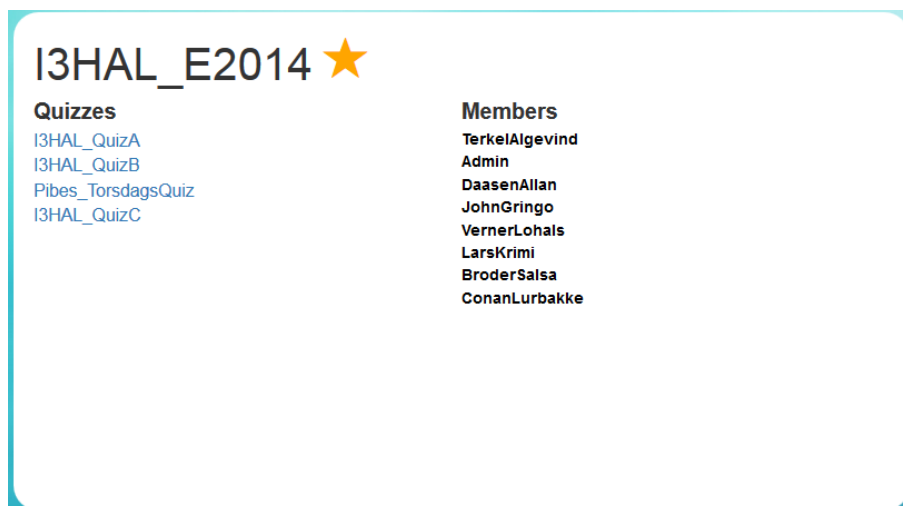


Figur 7.10. GUI til my Groups viewet

View Group

Dette view åbnes når man trykker på en gruppe i View My Groups viewet eller har fundet en gruppe med søgefunktionen.

Viewet er som vist på figur 7.11. Den viser to kolonner med de tilknyttede quizzer og medlemmerne i gruppen. Ved at trykke på quiznavnene åbnes Answer Quiz for den pågældende quiz. Det er også muligt at favorisere gruppen ved at klikke på stjernen ved siden af navnet på gruppen.



Figur 7.11. GUI til group viewet

7.3.3 User views

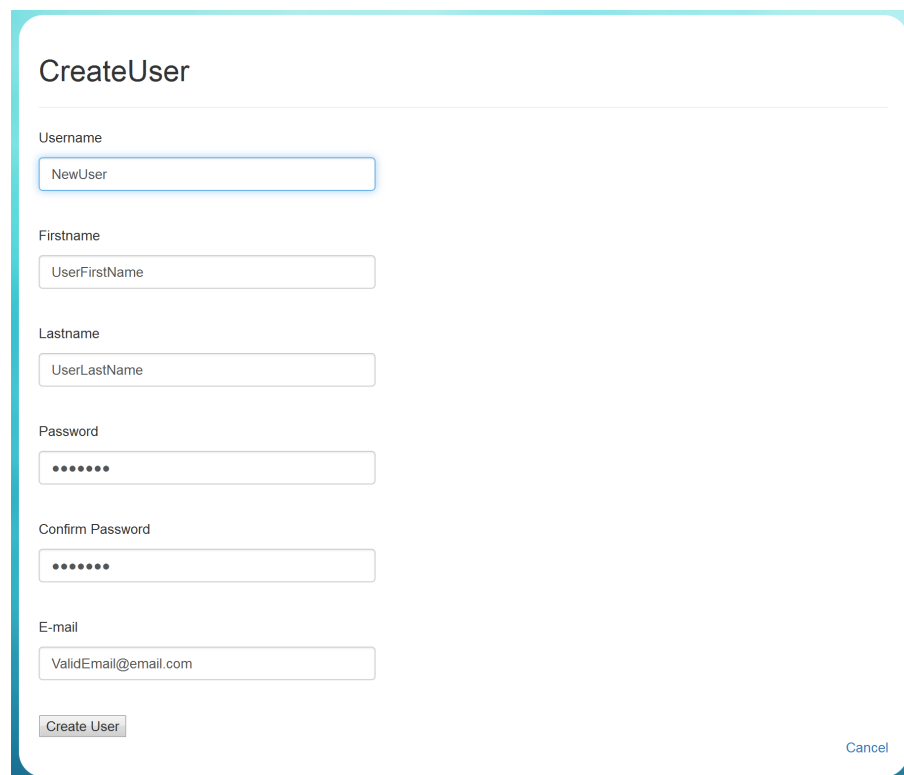
Til at håndtere brugerprofiler, har vi anvendt ASP.NET Identity-frameworket [Microsoft, 2015b]. ASP.NET Identity er et bibliotek, som har mange brugbare klasser, når det gælder håndtering af brugerprofiler. Klasser som gør det muligt at oprette en ny bruger med en bestemt adgangskode, slette en bruger, opdatere en bruger, ændre adgangskoden til en bruger, sende en E-mail og/eller SMS til en bruger og meget mere. Et argument for

at anvende ASP.NET Identity framework, er at det selv sørger for at hashe brugerens adgangskode, så denne ikke fremgår direkte i databasen.

Create User

Create User viewet kan tilgås ved at klikke på "Login" og derefter vælge "Register". Inde på Create User viewet kan en bruger oprette en ny brugerprofil. Viewet indeholder en række inputfelter, hvor brugeren skal indtaste og oplyse personlige informationer. Der er krav om at brugeren skal vælge et brugernavn og adgangskode samt oplyse en gyldig email-adresse. Derudover kan brugeren også oplyse fornavn og efternavn, men dette er ikke et krav. Til oprettelse af en bruger med en ny bestemt adgangskode anvendes `Create()`-metoden fra `UserManager`-klassen. Create metoden tager to parametre: Den nye brugerprofil, samt adgangskoden til den pågældende brugerprofil.

Figur 7.12 viser Create User viewet.

The image shows a web form titled "CreateUser". It contains several input fields: "Username" with the placeholder text "NewUser", "Firstname" with "UserFirstName", "Lastname" with "UserLastName", "Password" and "Confirm Password" both with placeholder dots, and "E-mail" with "ValidEmail@email.com". At the bottom left is a "Create User" button, and at the bottom right is a "Cancel" link.

Figur 7.12. Create User view

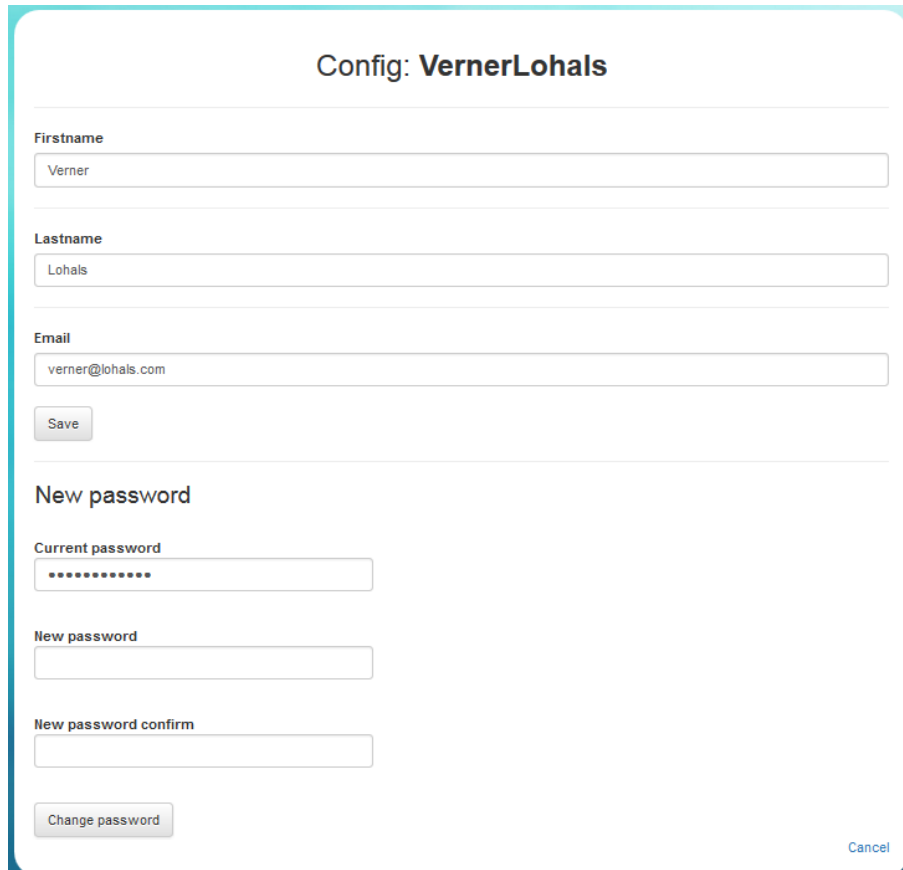
Config User

En bruger med en brugerprofil kan til hver en tid konfigurere sin profils informationer. Dog er det ikke muligt for en bruger at ændre sit brugernavn. Grunden til dette er, at de quizzer og grupper, som en bruger opretter, refererer til den pågældende brugers brugernavn. Det betyder altså, at når en bruger opretter en ny brugerprofil, vil det pågældende brugernavn være lås til brugeren.

Hvis en bruger ønsker at ændre sin adgangskode, skal brugeren oplyse den nuværende aktive adgangskode og derefter indtaste en ny adgangskode. For at ændre en brugers adgangs-

kode anvendes `ChangePassword()`-metoden fra `UserManager`-klassen. `ChangePassword()`-metoden tager tre parametre, bruger ID, nuværende adgangskode og den nye adgangskode. Men før der ændres på adgangskoden, skal der først undersøges om brugerens input af den nuværende adgangskode er gyldig. Til det anvendes `VerifyHashedPassword()`-metoden, som også er en del af `UserManager`-klassen. `VerifyHashedPassword()` hasher brugerens input af den nuværende adgangskode og sammenligner det med den nuværende originale hashede adgangskode.

Figur 7.13 viser Config User viewets opbygning.

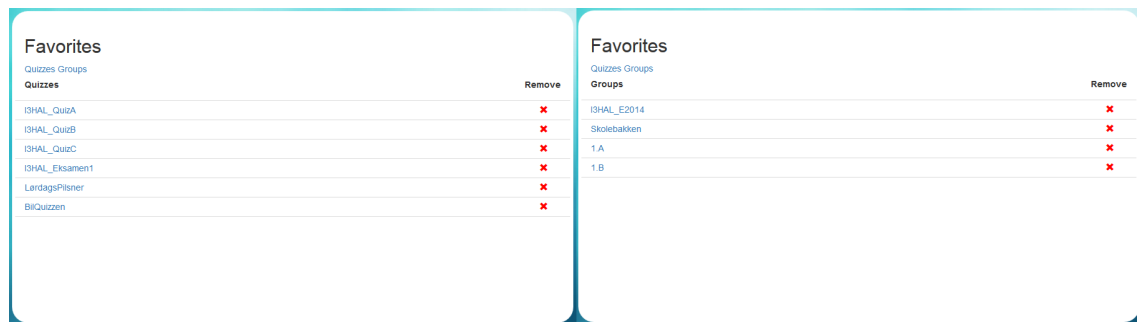


Figur 7.13. GUI af Config User

7.3.4 Favorite views

View Favorites

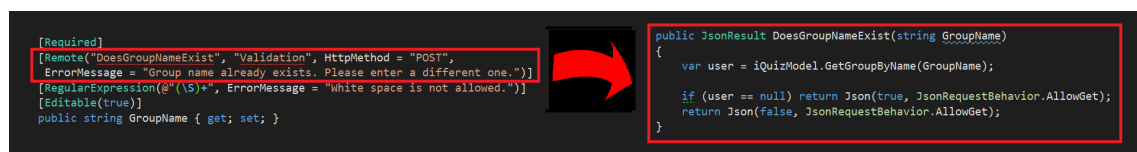
Favorites viewet er brugerens hurtig-adgang til quizzer og grupper. Brugeren kan tilføje favoritgrupper og -quizzer fra andre views, og derefter kunne tilgå disse i dette view. Dette kan være favorabelt frem for at søge efter quizzer og grupper hver gang de skal tilgås. Viewet deles op i to partial views: Æt for favoritgrupper og et for favoritquizzer, hvor favoritgrupper er det partialview, som vises når viewet åbnes via menuknappen. Viewet med sine to partialviews ses i figur 7.14. Quiz- og gruppenavnene agerer som links til hhv. Answer Quiz og View Group views. Favoritquizzer og -grupper kan fjernes ved at trykke på knappen i "remove" kolonnen.



Figur 7.14. GUI af Favorites viewet: quizzes til venstre, grupper til højre

7.4 Remote Validation

Der er implementeret Remote Validation til at finde ud af om gruppe- og brugernavne er taget eller ej. Dette fungerer ved at viewet via model-binding kigger på Groupname- og Username attributterne og ser at der er opsat remote validation til attributterne. Der bliver kaldt en validation metode i RemoteValidationControlleren som så finder ud af om det navn brugeren har indtastet, allerede er taget eller ej. Hvis navnet er ledigt returneres true og intet udskrives i viewet. Men hvis navnet er optaget returneres der false og den givne fejlbesked bliver udskrevet. Eksempel på en ErrorMessage kan ses på figur 7.15



Figur 7.15. Viser Remote Validation notation på modellen og funktionen den kalder

På figur 7.15 ses der implementeringen af remote validering til group name. [Remote ...] ligger i System.Web.Mvc.RemoteAttribute namespace og er det som der kalder valideringsmetoden DoesGroupNameExist.

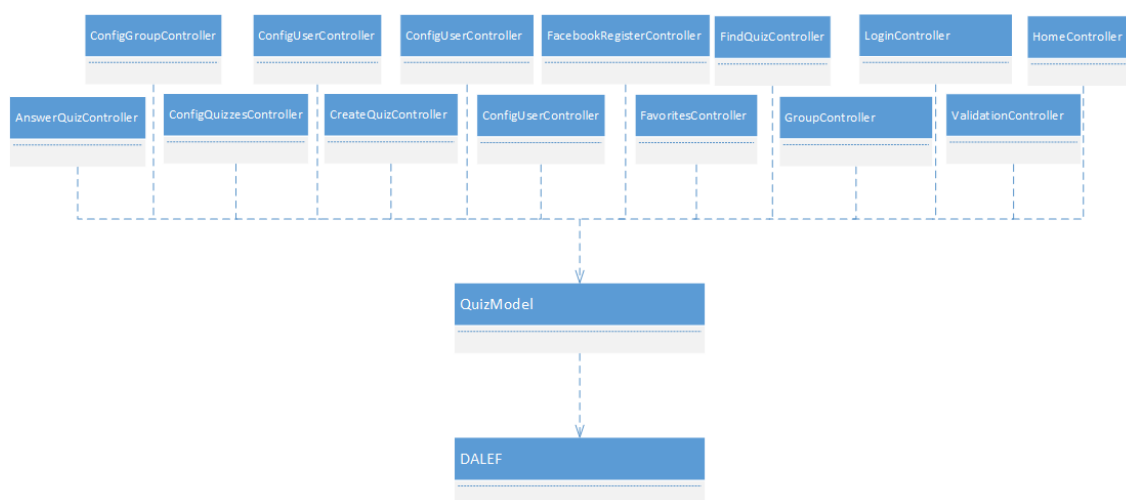
Sideløbende med applikationens udvikling, er der udviklet en række unit- og integrations-tests til at sikre funktionaliteten er som forventet. Begrundelsen for dette er at applikationen udvikles agilt, hvilket betyder at der kommer tilføjelser til eksisterende kode over hele projektets levetid. For at sikre at nye tilføjelser ikke ændrer eller ødelægger eksisterende kode.

Yderligere er anvendt en CI-Server. Denne er sat op til at bygge og teste projektet ved hvert commit af kode. På denne måde sikrer man at koden er uafhængig af den enkelte programmørs udviklingsmiljø, og får samtidig en række værktøjer til at overvåge fremgangen i applikationens udviklingen.

ASE har stillet en server til rådighed med CI-programmet Jenkins [Kawaguchi, 2015]. Adgang til denne er på URL: <http://10.29.0.31:8080/job/I4PRJ4G3/>. Bemærk at man skal være logget på ASEs netværk enten direkte eller med VPN for at tilgå serveren.

Til at strukturere og effektivisere unit- og integrationstestene er anvendt NUnit [Poole et al., 2014] og NSubstitute [Egerton and Tchepak, 2014]. Sammen med unittestene har vi anvendt dotCover [JetBrains, 2014] til at lave coverage analysis. Dette er et redskab som sikrer at alle forgreninger i koden rammes i ens tests.

Integrationstestene er lavet ud fra Collaboration-konceptet, hvor der tages udgangspunkt i use case-beskrivelserne. Her testes det at de enkelte moduler har den korrekte funktionalitet. For at udvælge de nødvendige tests udvikles et dependency-diagram der viser de enkelte klassers afhængigheder. På figur 8.1 er diagrammet for applikationen vist.



Figur 8.1. Dependency diagram for applikationen der viser klassernes afhængigheder

9

Accepttest-specifikation

For den generelle input test gælder minimum og maksimum karakterlængder er som beskrevet i de ikke-funktionelle krav, afsnit 3.3, på side 19.

0: Generel Input Test		
1	Test	1. Marker tekstboks. 2. Indtast minimum antal A karakterer minus 1. 3. Fjern markering fra tekstboks.
	Forventet resultat	Der udskrives fejlbesked
	Resultat	-
	Godkendt / Kommentar	-
2	Test	1. Marker tekstboks. 2. Indtast minimum antal A karakterer. 3. Fjern markering fra tekstboks.
	Forventet resultat	Der udskrives ingen fejlbesked
	Resultat	-
	Godkendt / Kommentar	-
3	Test	1. Marker tekstboks. 2. Indtast maksimalt antal A karakterer. 3. Fjern markering fra tekstboks.
	Forventet resultat	Der udskrives ingen fejlbesked
	Resultat	-
	Godkendt / Kommentar	-

...fortsat fra forrige side

0: Generel Input Test		
4	Test	<ol style="list-style-type: none"> 1. Marker tekstboks. 2. Indtast maksimalt antal A plus 1. 3. Fjern markering fra tekstboks.
	Forventet resultat	Der udskrives fejlbesked
	Resultat	-
	Godkendt / Kommentar	-

UC1: Create Quiz		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten for UC4: Create User test 1, for at oprette en bruger. 2. Udfør accepttesten UC6: Login test 1 for at logge ind. 3. Tryk på "Quizzes" og derefter "Create new quiz". 4. Udfør alle tests i Test 0: Generel InputTest Generel InputTest udføres på de følgende tekstbokse: <ul style="list-style-type: none"> Quizname Tags Question 1 Answer 1 Answer 2
	Forventet resultat	Alle interne tests i Test 0 er gennemført og godkendt
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC1: Create Quiz		
2	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten for UC4: Create User test 1 for at oprette en user. 2. Udfør accepttesten UC6: Login test 1 for at logge ind. 3. Tryk på "Quizzes" og derefter vælg "Create new quiz". 4. Indtast følgende værdier: Quizname: TestQuiz. Tags: tag1 tag2 Question 1: Spørgsmål 1 Answer 1: Rigtigt svar med check i boxen "Correct answer" og angiver points til 10. Answer 2: Forkert svar uden check i boxen "Correct answer" og angiver points til 0. 5. Tryk på "2" i liste over spørgsmål og vælg "Remove question" 6. Der trykkes på "Create Quiz".
	Forventet resultat	Quiz med 1 spørgsmål indeholdende 2 svarmuligheder med de indtastede navne og værdier er oprettet i databasen på database serveren.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC1: Create Quiz		
3	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten for UC4: Create User test 1 for at oprette en user. 2. Udfør accepttesten UC6: Login test 1 for at logge ind. 3. Tryk på "Quizzes" og derefter vælg "Create new quiz". 4. Indtast følgende værdier: <p>Quizname: TestQuiz2.</p> <p>Tags: tag2, tag3</p> <p>Question 1: Spørgsmål 1</p> <p>Answer 1: Rigtigt svar med check i boxen "Correct answer" og angiver points til 10.</p> <p>Answer 2: Forkert svar uden check i boxen "Correct answer" og angiver points til -5.</p> 5. Der trykkes på "Add new answer", og følgende værdier indtastes: <p>Answer 3: Semirigtigt svar med check i boxen "Correct answer" og angiver points til 5.</p> 6. Der trykkes på "Gennemse..." og en billedfil med navnet billede.jpg vælges. 7. Tryk på "2" i liste over spørgsmål og vælg "Remove question". 8. Der trykkes på "Create Quiz".
	Forventet resultat	Quiz med 1 spørgsmål indeholdende 3 svarmuligheder med de indtastede navne og værdier er oprettet i databasen på database serveren samt stien til billedfilen. Billedfilen er uploadet til webserveren
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC2: Search		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC1: Create Quiz test 2. 2. Aktiver søgefeltet. 3. Indtast TestQuiz og tryk på søg.
	Forventet resultat	Quizen TestQuiz vises som søgeresultat.
	Resultat	-
	Godkendt / Kommentar	-

...fortsat fra forrige side

UC2: Search		
2	Test	1. Udfør accepttesten UC1: Create Quiz test 2 og test 3. 2. Aktiver søgefeltet. 3. Indtast tag2 og tryk på søg.
	Forventet resultat	Quizzene TestQuiz og TestQuiz2 vises som søgeresultat.
	Resultat	-
	Godkendt / Kommentar	-
3	Test	1. Udfør accepttesten UC9: Create Group test 1. 2. Aktiver søgefeltet. 3. Indtast TestGroupNavn1 og tryk på søg. 4. Tryk på "Groups"
	Forventet resultat	Gruppen TestGroupNavn1 vises som søgeresultat.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
4	Test	1. Udfør accepttesten UC9: Create Group test 1. 2. Aktiver søgefeltet. 3. Indtast tag1 og tryk på søg. 4. Tryk på "Groups"
	Forventet resultat	Gruppene TestGroupNavn1 vises som søgeresultat.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
5	Test	1. Udfør accepttesten UC1: Create Quiz test 2. 2. Aktiver søgefeltet. 3. Indtast AAA og tryk på søg.
	Forventet resultat	Der vises ingen quiz i søgeresultaterne.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC2: Search		
6	Test	1. Udfør accepttesten UC1: Create Quiz test 2. 2. Aktiver søgefeltet. 3. Tryk på søg uden at indtaste noget i søgefeltet.
	Forventet resultat	Der sker intet.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC3: Answer Quiz		
1	Test	1. Udfør accepttesten UC2: Search test 1. 2. Vælg Quizzen TestQuiz . 3. Ved "Spørgsmål 1" vælges svaret "Rigtigt svar". 4. Tryk på "Finish quiz".
	Forventet resultat	Det vises, at der er svaret korrekt på 1 ud af 1 spørgsmål, og der er opnået 10 point.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	1. Udfør accepttesten UC2: Search test 1. 2. Vælg Quizzen TestQuiz . 3. Ved "Spørgsmål 1" vælges svaret "Forkert svar". 4. Tryk på "Finish quiz".
	Forventet resultat	Det vises, at der er svaret korrekt på 0 ud af 1 spørgsmål, og der er opnået 0 point.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
3	Test	1. Udfør accepttesten UC2: Search test 2. 2. Vælg Quizzen TestQuiz2
	Forventet resultat	Ved "Spørgsmål 1" ses billedet som i accepttesten for UC1: Create Quiz test 3 blev uploaded.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC3: Answer Quiz		
4	Test	1. Udfør accepttesten UC2: Search test 2. 2. Vælg Quizzen TestQuiz2 3. Ved "Spørgsmål 1" vælges svaret "Semirigtigt svar". 4. Tryk på "Finish quiz".
	Forventet resultat	Det vises, at der er svaret korrekt på 1 ud af 1 spørgsmål, og der er opnået 5 point.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
5	Test	1. Udfør accepttesten UC2: Search test 2. 2. Vælg Quizzen TestQuiz2 3. Ved "Spørgsmål 1" vælges svaret "Forkert svar". 4. Tryk på "Finish quiz".
	Forventet resultat	Det vises, at der er svaret korrekt på 0 ud af 1 spørgsmål, og der er opnået -5 point.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC4: Create User		
1	Test	1. Tryk på "Login" og vælg "Register" på brugergrænsefladen. 2. Indtast følgende værdier: Username: Testbruger1 . Email: testmail@test.dk . Password: password . Re-enter Password: password 3. Tryk på "Create User"
	Forventet resultat	Brugeren oprettes i databasen
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC4: Create User		
2	Test	<ol style="list-style-type: none"> 1. Udfør test 1 2. Tryk på "Login" og vælg "Register" på brugergrænsefladen. 3. Indtast følgende værdier: <ul style="list-style-type: none"> Username: Testbruger1. Email: testmail2@test.dk. Password: password. Re-enter Password: password 4. Tryk på "Create User"
	Forventet resultat	Der gives besked at en bruger med det ønskede brugernavn allerede findes i systemet. Brugeren oprettes ikke.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	<ol style="list-style-type: none"> 1. Udfør test 1 2. Tryk på "Login" og vælg "Register" på brugergrænsefladen. 3. Indtast følgende værdier: <ul style="list-style-type: none"> Username: Testbruger2. Email: testmail@test.dk. Password: password. Re-enter Password: password 4. Tryk på "Create User"
	Forventet resultat	Der gives besked at en bruger med samme email allerede findes i systemet. Brugeren oprettes ikke.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC5: Config User		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttest for UC4: Create User test 1 for at oprette en user. 2. Udfør accepttest for UC6: Login test 1 3. Tryk på "Login"og vælg "Account". 4. Ændre "Fornavn", "Efternavn"og "Email": Fornavn: FornavnUser Efternavn: EfternavnUser Email: Nytestmail2@test.dk 5. Tryk på "Save".
	Forventet resultat	Brugeren har ændret sit Fornavn, Efternavn og Email.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	<ol style="list-style-type: none"> 1. Udfør test 1 2. Tryk på "Login"og vælg "Account". 3. Skift Password: Current Password: abcdef New Password: NyPassword New PasswordConfirm: NyPassword 4. Tryk på "Change".
	Forventet resultat	Bruger bliver informeret om at den nuværende password ikke er gyldigt.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
3	Test	<ol style="list-style-type: none"> 1. Udfør test 1 2. Tryk på "Login"og vælg "Account". 3. Skift Password: Current Password: password New Password: NyPassword New PasswordConfirm: NyPassword 4. Tryk på "Change".
	Forventet resultat	Bruger bliver informeret om at Password er ændret.
	Resultat	Som forventet

...fortsat fra forrige side

UC5: Config User		
	Godkendt / Kommentar	Godkendt

6: Login		
1	Test	1. Brugeren klikker på "Login" 2. Udfør accepttesten UC4: Create user 3. Brugeren indtaster Email, password og trykker på sign in
	Forventet resultat	Login knappen er opdateret og fremstår nu som brugerens valgte display navn (email eller username). Brugeren er nu logget ind
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	1. Brugerne klikker på "Login" 2. Brugeren klikker på Facebook-knappen "Log in"
	Forventet resultat	Login knappen er opdateret og fremstår nu som brugerens valgte display navn (email eller brugernavn). Brugeren er nu logget ind
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

7: Logout		
1	Test	1. Udfør accepttesten UC6: Login 2. Brugeren klikker på display navnet som identificere brugerens account. 3. Brugeren klikker på "Logout"
	Forventet resultat	Siden opdateres og hvor display navnet før stod, står der nu "Login". Brugeren er logget ud
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

8: ConfigQuizzes		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login 2. Udfør accepttesten UC1: Create Quiz test 2 3. Bruger klikker på "Quizzes" 4. Bruger trykker på blyant logoet ud fra "TestQuiz" 5. Tilføj "1" efter den eksisterende tekst i alle felter 6. Klik på "Save" 7. Klik på "TestQuiz1"
	Forventet resultat	Indholdet af quizzen præsenteres med nyt indhold i alle felter
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login 2. Udfør accepttesten UC1: Create Quiz test 2 3. Bruger klikker på "Quizzes" 4. Bruger trykker på blyant logoet ud fra "TestQuiz" 5. Udfør alle tests i Test 0: Generel InputTest for felterne: <ul style="list-style-type: none"> QuizName Tags Question 1 Answer 1 Answer 2
	Forventet resultat	Alle interne tests i Test 0 er gennemført og godkendt
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
3	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login 2. Udfør accepttesten UC1: Create Quiz test 2 3. Bruger klikker på "Quizzes" 4. Bruger trykker på slet ved siden af quiznavn "TestQuiz" 5. Bekræft sletning 6. Indtast "TestQuiz" i søgefeltet
	Forventet resultat	Ingen resultater fundet
	Resultat	Som forventet

...fortsat fra forrige side

8: ConfigQuizzes		
	Godkendt / Kommentar	Godkendt

UC9: Create Group		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login test 1 2. Udfør accepttesten UC1: Create Quiz 3. Tryk på "Groups" knappen i menuen. 4. Tryk på knappen "Create new group". 5. Udfør accepttesten UC0: General Input Test 6. Indtast følgende værdier: <p style="margin-left: 40px;">Group Name: TestGroupNavn1</p> <p style="margin-left: 40px;">Tags: tag1 tag2</p> 7. Ud for TestQuiz sættes der et flueben for at tilføje quizzen til gruppen. 8. Tryk på "Create".
	Forventet resultat	Gruppen TestGroupNavn1 er oprettet i databasen på database serveren.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login test 1 2. Tryk på "Groups" knappen i menuen. 3. Tryk på knappen "Create new group". 4. Indtast følgende værdier: <p style="margin-left: 40px;">Group Name: TestGroupNavn2</p> <p style="margin-left: 40px;">Tags: tag1 tag2</p> 5. Tryk på "Create".
	Forventet resultat	Gruppen TestGroupNavn2 er oprettet i databasen på database serveren.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC9: Create Group		
3	Test	<ol style="list-style-type: none"> 1. Udfør test 2 2. Tryk på "Group" knappen i menuen. 3. Tryk på knappen "Create new group". 4. Indtast følgende værdier: Group Name: TestGroupNavn2
	Forventet resultat	Der vises en fejlmeddelelse, da gruppens navn allerede findes i databasen
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
4	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login test 1 2. Tryk på "Groups" knappen i menuen. 3. Tryk på knappen "Create new group". 4. Indtast følgende værdier: Group Name: TestGroupNavn3 Tags: tag1 tag2 5. Tryk på linket "Back".
	Forventet resultat	Gruppen TestGroupNavn3 oprettes ikke i databasen på database serveren. Brugerens grupper vises igen.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC10: Config Group		
1	Test	<ol style="list-style-type: none"> 1. Udfør accepttest for UC6: Login test 1 2. Udfør accepttest for UC9: Create Group test 1 3. Udfør accepttest for UC1: Create Quiz test 2 4. Klik på "Groups". 5. Vælg at konfigurere "TestGroupNavn1". 6. Vælg at slette "TestGroupNavn1". 7. Gruppensiden opdateres
	Forventet resultat	"TestGroupNavn1" er blevet slettet fra gruppen.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

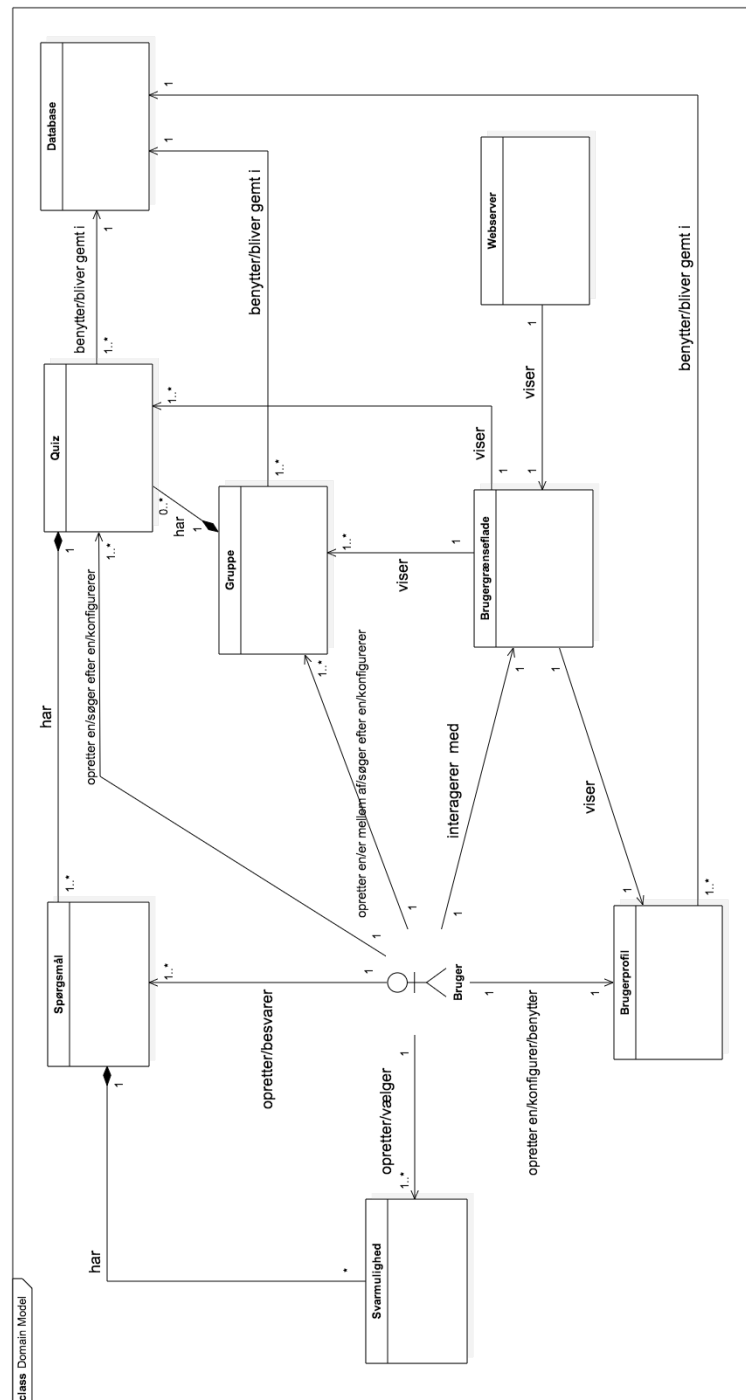
UC10: Config Group		
2	Test	<ol style="list-style-type: none"> 1. Udfør accepttest for UC6: Login test 1 2. Udfør accepttest for UC9: Create Group test 1 3. Udfør accepttest for UC1: Create Quiz test 2 4. Udfør accepttest for UC2: Search test 1 5. Klik på "Add to group" og vælg "TestGroupNavn1". 6. Klik på "Groups" 7. Vælg "TestGroupNavn1" gruppen.
	Forventet resultat	Bruger har tilføjet "TestQuiz" til gruppen.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
3	Test	<ol style="list-style-type: none"> 1. Udfør test 2. 2. Klik på blyanten ud for "TestGroupNavn1". 3. Sæt hak i feltet ud for "TestQuiz" og tryk "Update group"
	Forventet resultat	"TestQuiz" vises i listen af quizzer der kan tilføjes.
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
5	Test	<ol style="list-style-type: none"> 1. Udfør test 1. 2. Klik på blyanten ud for "TestGroupNavn1" 3. Tilføj "tag3" i feltet "Tags". 4. Klik "Update group" 5. Søg på "tag3" i søgefeltet 6. Klik på "Groups"
	Forventet resultat	"TestGroupNavn1" fremgår i resultaterne med "tag3"
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

UC11: Config Favorites		
0.1	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login test 1 for at logge ind. 2. Udfør accepttesten UC2: Search test 1, som søger efter oprettet Quiz. 3. Tryk på favorit-ikonet til venstre for quizzens navn for at tilføje denne til brugerens favoritter.
	Forventet resultat	Quizzen TestQuiz er gemt i databasen som favorit til brugeren.
	Resultat	-
	Godkendt / Kommentar	-
0.2	Test	<ol style="list-style-type: none"> 1. Udfør accepttesten UC6: Login test 1 for at logge ind. 2. Udfør accepttesten UC2: Search test 3 for at søge efter oprettet gruppe. 3. Tryk på favorit-ikonet til venstre for gruppens navn for at tilføje denne til brugerens favoritter.
	Forventet resultat	Gruppen TestGroupNavn1 er gemt i databasen som favorit til brugeren.
	Resultat	-
	Godkendt / Kommentar	-
1	Test	<ol style="list-style-type: none"> 1. Udfør test 0.1 som tilføjer favorit-quiz. 2. Tryk på "Favorites" på brugergrænsefladen. 3. Tryk på "Quizzes" på brugergrænsefladen.
	Forventet resultat	Quizzen TestQuiz vises som favorit
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
2	Test	<ol style="list-style-type: none"> 1. Udfør test 1 2. Tryk på "Remove favorite" ikonet ud for quizen TestQuiz
	Forventet resultat	Quizzen TestQuiz vises ikke længere i favorit-listen og er fjernet fra brugerens favoritter i databasen
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

...fortsat fra forrige side

UC11: Config Favorites		
3	Test	1. Udfør accepttesten 0.2 som tilføjer en favorit-gruppe. 2. Tryk på "Favorites" på brugergrænsefladen.
	Forventet resultat	Gruppen TestGroupNavn1 vises som favorit
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt
4	Test	1. Udfør test 3. 2. Tryk på "Remove favorite" ikonet ud for gruppen X
	Forventet resultat	Gruppen TestGroupNavn1 vises ikke længere i favorit-listen og er fjernet fra brugerens favoritter i databasen
	Resultat	Som forventet
	Godkendt / Kommentar	Godkendt

På bilags-cden vil det være muligt at finde doxygen tilhørende vores kodebase, selve kodebasen, en pdf udgave af rapport og af dokumentationen, samt mødereferater. Doxygen er som beskrevet tidligere en række HTML filer, som vil være at finde i den respektive mappe. De andre dele vil også være at finde i respektive mapper, så det er nemt at navigere og finde de forskellige dele. På cd'en vil der i mappen "HtmlClassModel" kunne findes en Html-dokumentation for klassediagrammer i systemet.



Figur A.1. Domænemodel der viser de logiske blokke og relationer for systemet

Litteratur

- Anthony Egerton and David Tchepak. NSubstitute, 2014. URL <http://nsubstitute.github.io>. Accessed 24.05.2015.
- E. Gamma, R. Helm, , R. Johnson, and J. Vlissides. *Design Patterns*. Addison Wesley, 2005.
- JetBrains. dotCover, 2014. URL <https://www.jetbrains.com/dotcover/>. Accessed 24.05.2015.
- Kohsuke Kawaguchi. Jenkins CI, 2015. URL <https://jenkins-ci.org>. Accessed 24.05.2015.
- Philippe Kruchten. Architectural Blueprints—The “4+1” View Model of Software Architecture, 1995. URL <https://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf>. Accessed 22.05.2015.
- Microsoft. ASP.NET MVC, 2014. URL <http://www.asp.net/mvc>. Accessed 24.05.2015.
- Microsoft. Azure, 2015a. URL <http://azure.microsoft.com/>. Accessed 24.05.2015.
- Microsoft. Identity MSDN, 2015b. URL <https://msdn.microsoft.com/en-us/library/dn613290%28v=vs.108%29.aspx>. Accessed 25.05.2015.
- Microsoft. Identity, 2015c. URL <http://www.asp.net/identity>. Accessed 25.05.2015.
- Microsoft. SQL Server, 2015d. URL <http://www.microsoft.com/SQLServer>. Accessed 24.05.2015.
- Microsoft. UserManager class MSDN, 2015e. URL [https://msdn.microsoft.com/en-us/library/microsoft.aspnet.identity.usermanagerextensions\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/microsoft.aspnet.identity.usermanagerextensions(v=vs.108).aspx). Accessed 25.05.2015.
- Olivier Müller, Marc Delisle, and Loïc Chapeaux. phpMyAdmin, 2015. URL <http://www.phpmyadmin.net/>. Accessed 24.05.2015.
- Charlie Poole, Rob Prouse, and Simone Busoli. NUnit, 2014. URL <http://www.nunit.org>. Accessed 24.05.2015.
- Wikipedia. SOLID (object-oriented design), 2015. URL [http://en.wikipedia.org/wiki/SOLID_\(object-oriented_design\)](http://en.wikipedia.org/wiki/SOLID_(object-oriented_design)). Accessed 21.05.2015.